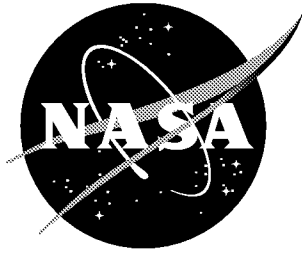


NASA/CR-2000-210313



Evaluation of Methods for Multidisciplinary Design Optimization (MDO), Part II

Srinivas Kodiyalam and Charles Yuan
Engineous Software, Inc., Morrisville, North Carolina

November 2000

The NASA STI Program Office ... in Profile

Since its founding, NASA has been dedicated to the advancement of aeronautics and space science. The NASA Scientific and Technical Information (STI) Program Office plays a key part in helping NASA maintain this important role.

The NASA STI Program Office is operated by Langley Research Center, the lead center for NASA's scientific and technical information. The NASA STI Program Office provides access to the NASA STI Database, the largest collection of aeronautical and space science STI in the world. The Program Office is also NASA's institutional mechanism for disseminating the results of its research and development activities. These results are published by NASA in the NASA STI Report Series, which includes the following report types:

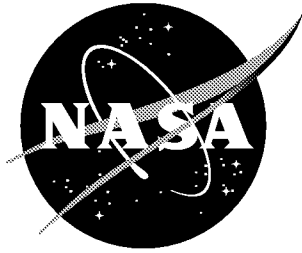
- **TECHNICAL PUBLICATION.** Reports of completed research or a major significant phase of research that present the results of NASA programs and include extensive data or theoretical analysis. Includes compilations of significant scientific and technical data and information deemed to be of continuing reference value. NASA counterpart of peer-reviewed formal professional papers, but having less stringent limitations on manuscript length and extent of graphic presentations.
- **TECHNICAL MEMORANDUM.** Scientific and technical findings that are preliminary or of specialized interest, e.g., quick release reports, working papers, and bibliographies that contain minimal annotation. Does not contain extensive analysis.
- **CONTRACTOR REPORT.** Scientific and technical findings by NASA-sponsored contractors and grantees.
- **CONFERENCE PUBLICATION.** Collected papers from scientific and technical conferences, symposia, seminars, or other meetings sponsored or co-sponsored by NASA.
- **SPECIAL PUBLICATION.** Scientific, technical, or historical information from NASA programs, projects, and missions, often concerned with subjects having substantial public interest.
- **TECHNICAL TRANSLATION.** English-language translations of foreign scientific and technical material pertinent to NASA's mission.

Specialized services that complement the STI Program Office's diverse offerings include creating custom thesauri, building customized databases, organizing and publishing research results ... even providing videos.

For more information about the NASA STI Program Office, see the following:

- Access the NASA STI Program Home Page at <http://www.sti.nasa.gov>
- E-mail your question via the Internet to help@sti.nasa.gov
- Fax your question to the NASA STI Help Desk at (301) 621-0134
- Phone the NASA STI Help Desk at (301) 621-0390
- Write to:
NASA STI Help Desk
NASA Center for AeroSpace Information
7121 Standard Drive
Hanover, MD 21076-1320

NASA/CR-2000-210313



Evaluation of Methods for Multidisciplinary Design Optimization (MDO), Part II

Srinivas Kodiyalam and Charles Yuan
Engineous Software, Inc., Morrisville, North Carolina

National Aeronautics and
Space Administration

Langley Research Center
Hampton, Virginia 23681-2199

Prepared for Langley Research Center
under Contract NAS5-96012

November 2000

Available from:

NASA Center for AeroSpace Information (CASI)
7121 Standard Drive
Hanover, MD 21076-1320
(301) 621-0390

National Technical Information Service (NTIS)
5285 Port Royal Road
Springfield, VA 22161-2171
(703) 605-6000

Table of Contents

1.0 Objectives	1
2.0 Recorded Work	1
3.0 MDO Methods	2
3.1 Bi-Level Integrated System Synthesis (BLISS) Procedure:	2
3.2 Bi-Level Integrated System Synthesis with Response Surfaces (BLISS/RS) Procedure:	8
3.3 Bi-Level Integrated System Synthesis for Structures (BLISS/S):	18
3.4 Automotive MDO Application using Massively Parallel Processing	37
3.5 Summary	43
4.0 References	44

Evaluation of Methods for Multidisciplinary Design Optimization (MDO), Part II

Srinivas Kodiyalam & Charles Yuan
Engineous Software, Inc.

Acknowledgement: The authors would like to acknowledge the technical guidance provided by Dr. Jaroslaw S. Sobieski of NASA Langley during the course of this project. His support was invaluable.

1.0 Objectives

The general objective of the MDO Method Evaluation project is to collect numerical data on a number of promising MDO methods as well as implementing and evaluating new methods with the intent of providing some practical guidelines for their use.

The objective of Phase I was to collect data on All-in-One Method (A-i-O, also referred to as Multidisciplinary Feasible Method (MDF)), Individual Discipline Feasible Method (IDF), and Collaborative Optimization (CO) (Kodiyalam, 1998, Alexandrov and Kodiyalam, 1998).

The objectives of Phases II to V, were to perform iSIGHT scripting language based implementation of the new MDO method, Bi-Level Integrated System Synthesis and its variants, BLISS, BLISS/RS, and BLISS/S. In addition, a task on testing a variant of BLISS/RS on a large scale, industrial MDO problem was performed using the 256 processor NASA Ames Origin 2000 machine.

2.0 Recorded Work

In this report, we record the work performed by each method during every optimization procedure.

For A-i-O, we report the total number of multidisciplinary analyses (MDA), including those necessary to compute the finite-difference derivatives. We also account the average number of fixed-point iterations taken to achieve each MDA. Thus, the average number of function evaluations for each run of A-i-O is equal to the number of MDA times the average number of fixed-point iterations per MDA times the number of disciplines.

For BLISS and its variants, we report the total number of the number of BLISS cycles, the total number of system analysis as well as the total number of all the subsystem/disciplinary analyses.

3.0 MDO Methods

3.1 Bi-Level Integrated System Synthesis (BLISS) Procedure:

BLISS is a recently introduced method (Sobieszczanski-Sobieski et al., 1998) that uses a gradient-guided path to reach the improved system design, alternating between the set of modular design subspaces (disciplinary problems) and the system level design space. BLISS is an A-i-O like method in that a complete system analysis is performed to maintain multidisciplinary feasibility at the beginning of each cycle of the path. However, the system level optimization problem with BLISS uses a relatively small number of design variables that are shared by the subspaces (disciplines) and solution of the system level problem is obtained using the derivatives of the behavior (state) variables with respect to system level design variables and the Lagrange multipliers of the constraints obtained at the solution of the disciplinary optimizations.

The BLISS procedure comprises of the system analysis and sensitivity analysis, local disciplinary optimizations, and the system optimization. The details of the complete BLISS procedure is provided in Sobieszczanski-Sobieski et al., 1998. For completeness, the key steps in the BLISS procedure are outlined below.

0. Initialize local disciplinary (X) and system level (Z) variables.
1. Perform system analysis (SA) to compute the state variables (Y) and the design constraint functions (G); the SA includes all the local disciplinary analysis (BBAs) for all the black boxes/disciplines (BBs).
2. Check for convergence of the BLISS procedure.
3. Perform black box sensitivity analysis (BBSA) to compute local derivatives, including, $d(Y,X)$, $d(Y_{r,s}, Y_r)$, $d(G,Z)$ and $d(G,Y)$; Perform system sensitivity analysis (SSA) to compute global derivatives $D(Y,X)$ and $D(Y,Z)$. Note that the subscript r refers to the r^{th} BB/discipline, Y_r corresponds to the vector of state variables output from BB_r , and $Y_{r,s}$ correspond to vector of variables input to BB_r from BB_s .
4. Black box (local disciplinary) optimization (BBOPT) for all the BBs to get ΔX_{opt} and the Lagrange multipliers (L) for the active constraints at the constrained optimum.
5. Perform optimal sensitivity analysis (OSA) to compute $D(\Phi, Z)$ for use with system optimization (SOPT) where, Φ is the SOPT objective function. Two different methods for computing the optimal sensitivities are outlined in reference, Sobieszczanski-Sobieski et al., 1998.
6. Solve SOPT to get ΔZ_{opt} .
7. Update X and Z and repeat from Step 1.

If the starting point is feasible, then the BLISS procedure will maintain feasibility while improving the system objective. Alternatively, if the starting point is infeasible, the constraint violations are reduced while minimizing the increase in system objective. A flow chart of the BLISS procedure is shown in Figure 3.1-1.

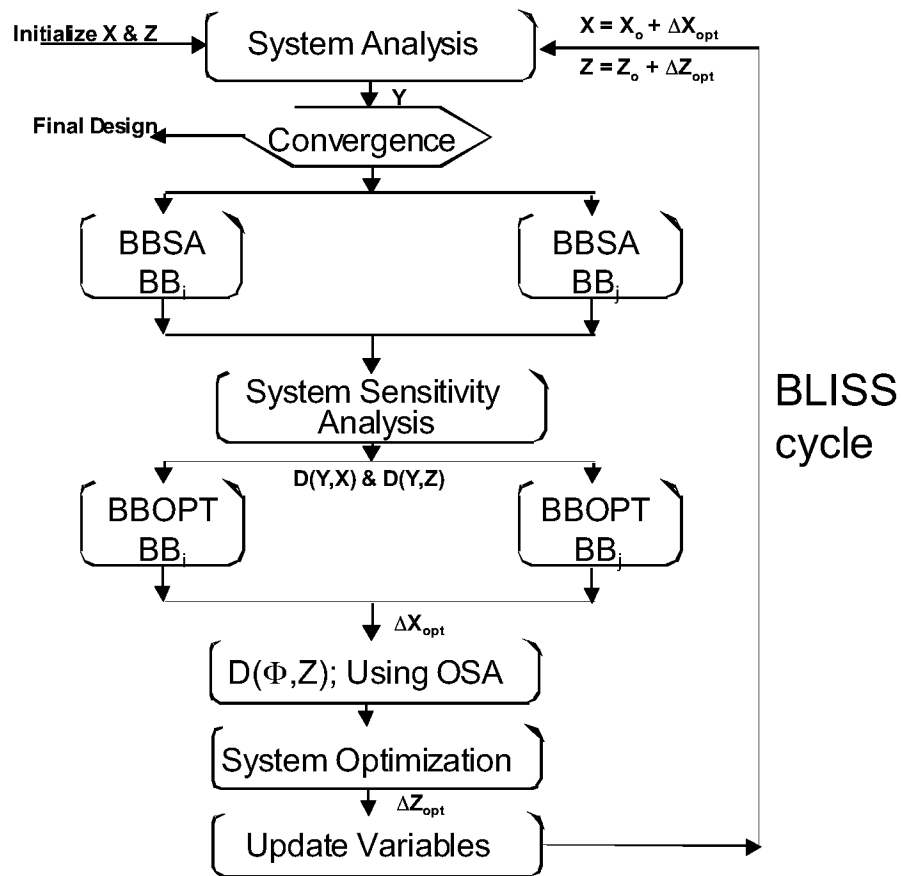


Figure 3.1-1 Flowchart of BLISS MDO method

3.1.1 BLISS MDO method – Numerical Examples and Results:

3.1.1.1 Example 1 - Electronic Packaging:

The electronic packaging is a multidisciplinary problem with coupling between electrical and thermal subsystems. Component resistance is influenced by operating temperatures; the temperatures depend on resistance. The objective of the problem is to maximize the watt density for the electronic package subject to constraints. The constraints require the operating temperatures for the resistors to be below a threshold temperature and the current through the two resistors to be equal. More details of the problem can be obtained in Renaud, 1993.

For the A-i-O approach, the optimization problem is given as follows:

Maximize: Y_1 (Watt Density)

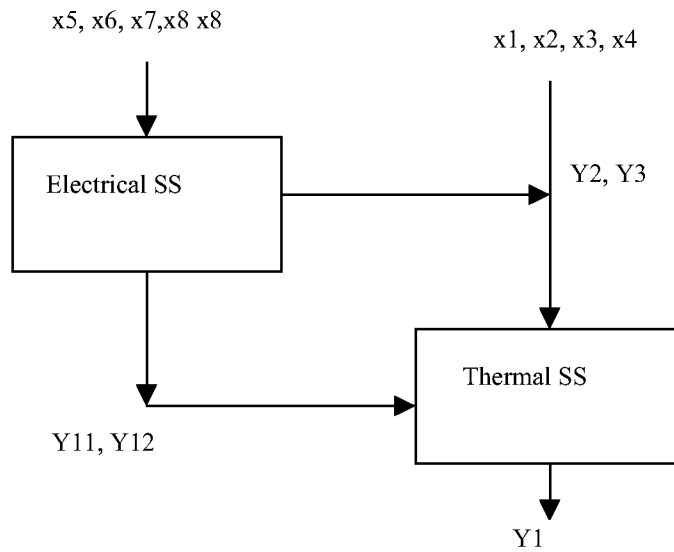
Subject to: $h_1 = Y_4 - Y_5 = 0.0$ (branch current equality)

$g_1 = Y_{11} - 85.0 \leq 0$ (component 1 reliability)

$g_2 = Y_{12} - 85.0 \leq 0$ (component 2 reliability)

The A-i-O problem has 8 design variables that are the following:

- $0.05 < \text{heat sink width } (x_1) \leq 0.15$
- $0.05 \leq \text{heat sink length } (x_2) \leq 0.15$
- $0.01 \leq \text{fin length } (x_3) \leq 0.10$
- $0.005 \leq \text{fin width } (x_4) \leq 0.05$
- $10.0 \leq \text{resistance \#1 } (x_5) \leq 1000.0$
- $0.004 \leq \text{temperature coefficient } (x_6) \leq 0.009$
- $10.0 \leq \text{resistance \#2 } (x_7) \leq 1000.0$
- $0.004 \leq \text{temperature coefficient } (x_8) \leq 0.009$



The system level objective function is to maximize Watt Density (Y_1). The system level optimization task has a total of 4 design variables (Z_2, Z_3, Z_{11}, Z_{12}) that are the coupling parameters between the 2 disciplines and physically represent the resistances and component temperatures.

The BLISS system optimization problem is stated as:

Find the set of system variables, \mathbf{Z} ,

Maximize: Y_1 (Watt Density)

Subject to: Bounds on \mathbf{Z}

The 2 subsystem optimization problems are stated as follows. The thermal subsystem optimization task is given as:

Maximize: $\phi = D(Y_1, X^1) \cdot \Delta X^1 + (Y_{11} - Z_{11})^2 + (Y_{12} - Z_{12})^2$

Subject to:

$$g_1 = Y_{11} - 85.0 \leq 0$$

$$g_2 = Y_{12} - 85.0 \leq 0$$

The thermal task has 4 design variables:
 X^1_i ; $i = 1,4$

The Electrical subsystem optimization task is given as:
 Maximize: $\phi = D(Y_1, X^2). \Delta X^2 + (Y_2 - Z_2)^2 + (Y_3 - Z_3)^2$
 Subject to: $h_1 = Y_4 - Y_5 = 0.0$

The Electrical task has 4 design variables:
 X^2_i ; $i = 5,8$

The Electronic Packaging problem was solved using iSIGHT for different starting points using the A-i-O and BLISS approaches.

Table 3.1.1-1: A-i-O Solutions

Case	Initial Design Objective	Initial Design Max Constraint Violation	Final Design Objective	Final Design Max Constraint Violation	Work
1	7.79440D+01	+2.16630D-08(3)	6.39720D+05	+1.21880D-03(3)	$83*3*2 = 498$
2	6.83630D+03	-2.89560D-01(3)	6.39720D+05	+1.21880D-03(3)	$44*3*2 = 264$
3	1.51110D+03	-4.29240D-02(3)	6.36540D+05	+1.45140D-03(3)	$44*3*2 = 264$
4	1.46070D+03	-1.02490D-03(3)	6.36940D+05	+1.42110D-03(3)	$35*3*2 = 210$

Table 3.1.1-2: BLISS Solutions using Abridged Algorithm for OSA

Case	Initial Design Objective	Initial Design Max Constraint Violation	Final Design Objective	Final Design Max Constraint Violation	Computational Effort	
					System analyses	Subsystem analyses
1	7.79440D+01	+2.16630D-08(3)	6.39720D+05	+1.22D-03(3)	9 (8 BLISS cycles)	365
2	6.83630D+03	-2.89560D-01(3)	6.39720D+05	+1.22D-03(3)	5 (4 BLISS cycles)	207
3	1.51110D+03	-4.29240D-02(3)	6.39720D+05	+1.22D-03(3)	3 (2 BLISS cycles)	114
4	1.46070D+03	-1.02490D-03(3)	6.39720D+05	+1.22D-03(3)	3 (2 BLISS cycles)	105

Table 3.1.1-3: BLISS Solutions using Sequential Linear Programming approach for OSA

Case	Initial Design Objective	Initial Design Max Constraint Violation	Final Design Objective	Final Design Max Constraint Violation	Computational Effort	
					System analyses	Subsystem analyses
1	7.79440D+01	+2.16630D-08(3)	6.39700D+05	+1.20D-03(3)	11 (10 BLISS cycles)	436
2	6.83630D+03	-2.89560D-01(3)	6.39050D+05	+1.18D-03(3)	14 (13 BLISS cycles)	508
3	1.51110D+03	-4.29240D-02(3)	6.39050D+05	-4.89D-04(3)	5 (4 BLISS cycles)	174
4	1.46070D+03	-1.02490D-03(3)	6.39290D+05	+3.70D-04(3)	9 (8 BLISS cycles)	313

The BLISS solutions using the abridged algorithm for OSA (Table 3.1.1-2) is consistently more efficient in terms of the total work required for convergence to the optimal solution.

Example 2: 3.1.1.2 Aircraft Optimization

In this example, a supersonic business jet modeled as a coupled system of structures (BB1), aerodynamics (BB2), propulsion (BB3), and aircraft range (BB4) is used. This problem is identical to the one used by Sobieszczanski-Sobieski et al., 1998, and complete details of the problem can be obtained from the same reference.

The mathematical formulation of the A-i-O optimization problem is as follows:

Maximize: Aircraft Range ($F(\mathbf{X})$)

Subject to constraints on:

Stress on wing < 1.09 ; ($G_j(\mathbf{X})$, $j=1,5$)

$0.96 < \text{Wing twist} < 1.04$; ($G_j(\mathbf{X})$, $j=6,7$)

Pressure gradient < 1.04 ; ($G_j(\mathbf{X})$, $j=8$)

$0.5 < \text{Engine Scale factor} < 1.5$; ($G_j(\mathbf{X})$, $j=9,10$)

Engine Temperature < 1.02 ; ($G_j(\mathbf{X})$, $j=11$)

Throttle setting $< T_{UA}$; ($G_j(\mathbf{X})$, $j=12$)

There are a total of 10 design variables, \mathbf{X} , including, thickness/chord ratio, altitude, Mach number, aspect ratio, wing sweep, wing surface area, taper ratio, wingbox cross-section, skin friction coefficient, and throttle. The A-i-O problem is solved using the Sequential Quadratic programming (DONLP) implementation in iSIGHT.

The BLISS decomposition consists of 4 subsystems including, Structures, Aerodynamics, Propulsion, and Range. A total of 6 system design variables are considered:

thickness/chord ratio, altitude, Mach number, aspect ratio, wing sweep, and wing surface area. BB1 (Structures) has two local variables (X_1 = taper ratio, wingbox cross-section), BB2 (Aerodynamics) has one local variable (X_2 = skin friction coefficient), and BB3 (Propulsion) has one local design variable (X_3 = throttle). BB4 computes the system objective Range and does not perform any local optimization.

The results obtained from BLISS method are compared with A-i-O method in Table 3.1.1-4.

CASE	Initial Objective	Initial Max. Constraint Value	Final Objective	Final Max. Constraint Value	Computational Effort	
					Number of System Analyses	Number of Subsystem Analyses
A-i-O	535.79	-0.162	3964.19	+1.0e-08	119	(119x4x3) 1428
BLISS	535.79	-0.162	3964.07	+1.92e-05	7 (6 BLISS cycles)	491

Table 3.2.4-1: Aircraft Optimization results using BLISS & comparison with A-i-O

3.2 Bi-Level Integrated System Synthesis with Response Surfaces

(BLISS/RS) Procedure:

The use of response surfaces with the BLISS method for system level optimization will (i) replace the need for subsystems' optimum sensitivity analysis (no $D(\Phi, Z)$ computations), and, (ii) eliminate the need for subsystem optimizations to yield a feasible solution (and, extrapolation issues concerning switching of active subsystem constraints) for each BLISS cycle (Kodiyalam and Sobieski, 1999). In addition, the smoothing operation resulting from the use of response surfaces may improve the convergence characteristics of the numerical optimization scheme, as well as reduce the possibility of being trapped in a local minimum.

In this work, the response surfaces are used only with the system optimization task and are constructed in the system design variables (Z) space. They are not used within the subsystem optimizations (BBOPT). Two algorithms, BLISS/RS1 and BLISS/RS2, that are modifications of the original BLISS outlined in the previous section are proposed. The primary difference between the two algorithms is that in BLISS/RS1 the response surfaces are constructed and updated using system analysis data (step 1 of BLISS procedure, Section 3.1) while in BLISS/RS2 the response surfaces are constructed using the subsystem (black box/disciplinary) optimization data (step 4 of BLISS procedure, Section 3.1) performed for linearly extrapolated Y variables. A flow chart of the BLISS procedure with response surfaces is shown in Figure 3.2-1.

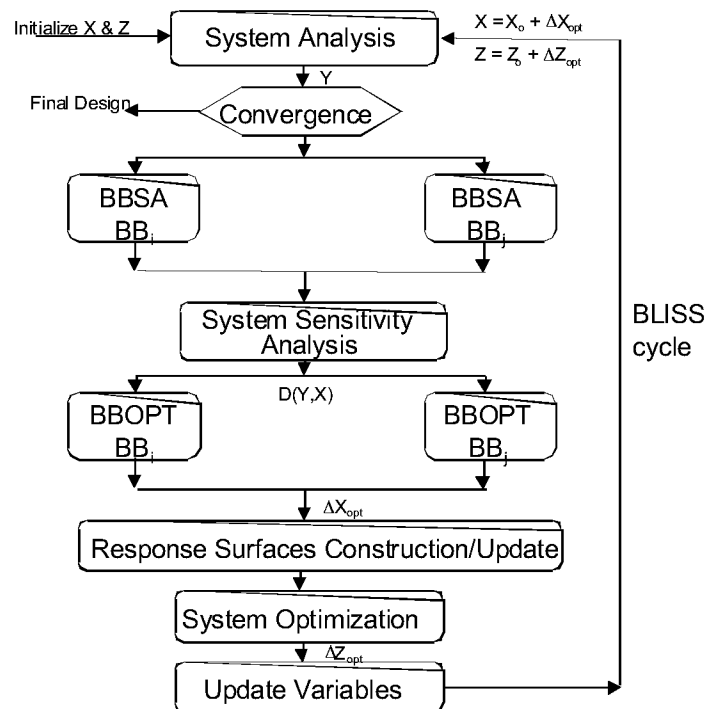


Figure 3.2-1. BLISS with Response Surfaces

3.2.1 BLISS/RS1: Algorithm 1

The BLISS/RS1 algorithm begins with a complete system analysis to compute the state variables (Y) and to ensure multidisciplinary compatibility at the start of each BLISS cycle. A local sensitivity analysis is then performed within each subspace/BB to compute the local sensitivities, $d(Y,X)$ and $d(Y_{r,s},Y_r)$. The local sensitivity analysis could be performed using an analytic, semi-analytic or finite difference method. The local sensitivities are then used with the system sensitivity analysis to formulate the global sensitivity equations (GSE). The solution of the GSE provide for the complete derivatives, $D(Y,X)$. The state variables computed from the system analysis and the global sensitivities are used with the solution of the r subspace optimization (BBOPT) problems for determining the changes (ΔX) in the local variables. The next stage in the BLISS/RS1 algorithm is the construction and/or the update of the response surfaces for the system level objective and constraint functions. The response surfaces are constructed in the Z (system variable) space by performing a complete system analysis for each Z_j generated randomly or using a experimental design (DOE) procedure. Having constructed the response surfaces, these are then used with the solution of the system optimization problem for determining the changes (ΔZ) in the system variables. Finally, the X and Z variables are updated and the BLISS cycle is repeated till a satisfactory convergence is obtained.

A step by step definition of the BLISS/RS1 algorithm is provided below.

0. Initialize local disciplinary (X) and system level (Z) variables.
1. System analysis (SA) to compute the state variables (Y) and the design constraint functions (G); the SA includes all the local disciplinary analysis (BBAs) for all the black boxes/disciplines (BBs).
2. Check for convergence of the BLISS procedure.
3. Black box sensitivity analysis (BBSA) to obtain $d(Y,X)$, and $d(Y_{r,s},Y_r)$; System sensitivity analysis (SSA) to compute $D(Y,X)$ only.
4. Maintain constant Z , and perform BBOPT in the X space for each BB. This BBOPT is the same as in the original BLISS procedure (outlined in Section 2.1).

Given X , Z from Step 7 and Y from SA in step 1:

Find ΔX that,

Minimizes $\phi = D(y_{r,i},X) \cdot \Delta X$

Satisfy $G(X,Y,Z) \leq 0$

Here, $y_{r,i}$ corresponds to an element of the vector Y_r . It is the system objective function that is computed as a single output item in one of the BBs (or, disciplines). The output is the optimal objective and X to be saved for use in step 7.

5.1 If this is the first pass (first cycle), operate in ΔZ space, generate $\Delta Z_j, j=1,N$ vectors using DOE methods or randomly, within reasonable move limits on ΔZ .

5.1.1 Perform SA (System analysis) for each $\Delta Z_j, j=1,N$, holding X not changed.

5.1.2 Generate response surfaces for the system objective (Φ) and each of the system constraints (G) in the Z space.

5.2 If this is the second or subsequent cycle:

5.2.1 Perform SA for X and Z updated in step 7

5.2.2 Optionally, add $\Delta Z_j, j=1,N$ vectors generated randomly or by a DOE method.

5.2.3 If any ΔZ_j were added in step 5.2.2, repeat SA for these ΔZ_j while holding X as updated in step 7.

5.2.4 Update the previously generated RS using the results from SA in step 5.2.1, and from step 5.2.3.

6. Given the response surfaces for Φ and special constraints G_{xz} from step 5.1.2 or updated in step 5.2.4, perform optimization in the Z -space:

Find ΔZ that,

Minimizes Φ

Satisfy $G_{xz} \leq 0$, and ΔZ within move limits.

Note that the special constraints, G_{xz} , are those constraints that are strongly dependent of both X and Z variables.

7. Update X and Z using the results from Step 4 and 6.

3.2.2 BLISS/RS2: Algorithm 2

A step by step definition of the BLISS/RS2 algorithm is provided below. As mentioned earlier, the primary difference between this algorithm and BLISS/RS1 is the procedure used for constructing the system objective and constraint response surfaces in the Z space. With BLISS/RS2, the response surfaces are constructed using the subspace optimization results performed for linearly extrapolated Y variables.

0. Initialize local disciplinary (X) and system level (Z) variables.

1. System analysis (SA) to compute the state variables (Y) and the design constraint functions (G); the SA includes all the local disciplinary analysis (BBAs) for all the black

boxes/disciplines (BBs).

2. Check for convergence of the BLISS procedure.

3. Black box sensitivity analysis (BBSA) to obtain $d(Y,X)$, $d(Y_{r,s},Y_r)$, $d(G,Z)$ and $d(G,Y)$; System sensitivity analysis (SSA) to compute $D(Y,X)$ and $D(Y,Z)$.

4. Black Box optimization and response surface update

4.1 If this is the first pass (first cycle), operate in ΔZ space, generate ΔZ_j , $j=1,N$ vectors using DOE methods or randomly, within reasonable move limits on ΔZ .

4.1.1 For each ΔZ_j , extrapolate $Y = Y$ (from Step 1) + $D(Y,Z).\Delta Z_j$

4.1.2 Perform BBOPT in the X space for each BBs that produces the objective function. This BBOPT is the same as in the original BLISS procedure (outlined in Section 2.1), except of Y being tied to Z through Step 4.2 above.

Given X (from Step 1), $Z = Z$ (from Step 1) + ΔZ_j (from Step 4.1), and $Y = Y(Z)$

(from Step 4.1.1):

Find ΔX that,

Minimizes $\phi = D(y_{r,i},X). \Delta X$

Satisfy $G(X,Y,Z) \leq 0$

As stated in the prior section, $y_{r,i}$ corresponds to an element of the vector Y_r and this corresponds to the system objective function that is computed as a single output item in one of the BBs (or, disciplines).

4.1.3 Generate response surfaces for each ϕ of each BB in the Z space and response surfaces for the special constraints G_{xz} that are direct functions of X and Z .

4.2 If this is the second or subsequent pass (cycle):

4.2.1 Repeat step 4.1.1 and 4.1.2 only for the single ΔZ obtained in step 5 or, optionally, for additional ΔZ vectors generated randomly or by a DOE method.

4.2.2 Update previously generated response surfaces for each ϕ and G_{xz} to accommodate new data corresponding to the design points corresponding to the ΔZ s used in step 4.2.1.

5. Given the response surfaces for ϕ and special constraints G_{xz} for each BB, obtained in step 4.1.3 in the first cycle, or from step 4.2.2 in subsequent cycles

Find ΔZ that,

Minimizes $\sum_i (\Delta\phi \text{ from BB}_i); i = 1 \dots \text{all BB's}$

Satisfy $G_{xz} \leq 0$, and ΔZ within move limits.

6. Update X and Z and begin next cycle from Step 1.

Needless to mention, the quality of the response surfaces is critical to improving the computational efficiency of the BLISS procedure (alternatively, reducing the number of BLISS cycles). The actual procedure of the response surface construction is outlined in Section 3.2.3

3.2.3 Response Surface Construction

In this work, an “adaptable” response surface model (RSM) implementation in iSIGHT software is used (Golovidov, Kodiyalam et al., 1998). In this approach, a minimum number of designs are used to construct an initial model around the baseline design. Typically, a linear model is constructed initially, although the user has an option to request a quadratic initial model. For a linear model, this number would be $(N_{inp}+1)$, where N_{inp} is the number of inputs. After the best design is found using this model within the specified design space bounds, the design is analyzed using the "Exact analysis", the data is included into the model data set, and the model is regenerated. The cycle is repeated with new design space bounds and the model is updated with another optimum design for the current model state. Each additional design in the model data set allows for the definition of one additional quadratic term in the polynomial, up to a full quadratic, after which a least squares fit is used for calculating the coefficients. Since the initial designs constitute only a small fraction of the total data set of the model, their effect is diminished and their distribution in the design space is of much less importance than in the case when all designs for model construction are distributed and analyzed up front. iSIGHT uses randomly generated or DOE generated designs for the initial model. The described approach allows the model to be built at run time following the path of the optimizer, and automatically provides more designs for the model near the region of the optimum, resulting in the increased accuracy of the model near the optimum design. In most simple problems convergence occurs before a full quadratic polynomial is constructed or soon thereafter. In more complicated problems with functions of non-trivial shape, restarting of optimization and regenerating of the response surface model may still be required. The algorithm proved to be very efficient and reliable and was tested on several realistic design problems.

The order in which the quadratic coefficients of the model polynomial are defined is determined by the order of input parameters of the model. As more and more design points become available, diagonal quadratic terms are first calculated, and then mixed coefficients are defined. The RSM performance can be improved by using the results of a DOE study and analysis of variance (ANOVA) to determine the most important input parameters, and then use that information for setting the order of defining the model coefficients (Kodiyalam et al., 1998).

3.2.4 BLISS/RS Numerical Examples

Two design examples are used to test and demonstrate the BLISS procedure with response surfaces. Both the aircraft design optimization (Sobieszczanski-Sobieski et al., 1998) and the conceptual ship design problem (Kodiyalam et al., 1997) use low fidelity analysis codes representative of a conceptual design stage. The results from the BLISS/RS1 and BLISS/RS2 methods are compared with the conventional A-i-O and original BLISS methods.

3.2.4.1 Aircraft Optimization

In this example, a supersonic business jet modeled as a coupled system of structures (BB1), aerodynamics (BB2), propulsion (BB3), and aircraft range (BB4) is used. This problem is identical to the one used by Sobieszczanski-Sobieski et al., 1998, and complete details of the problem can be obtained from the same reference.

The mathematical formulation of the A-i-O optimization problem is as follows:

Maximize: Aircraft Range ($F(\mathbf{X})$)

Subject to constraints on:

Stress on wing < 1.09 ; ($G_j(\mathbf{X})$, $j=1,5$)

$0.96 < \text{Wing twist} < 1.04$; ($G_j(\mathbf{X})$, $j=6,7$)

Pressure gradient < 1.04 ; ($G_j(\mathbf{X})$, $j=8$)

$0.5 < \text{Engine Scale factor} < 1.5$; ($G_j(\mathbf{X})$, $j=9,10$)

Engine Temperature < 1.02 ; ($G_j(\mathbf{X})$, $j=11$)

Throttle setting $< T_{UA}$; ($G_j(\mathbf{X})$, $j=12$)

There are a total of 10 design variables, \mathbf{X} , including, thickness/chord ratio, altitude, Mach number, aspect ratio, wing sweep, wing surface area, taper ratio, wingbox cross-section, skin friction coefficient, and throttle. The A-i-O problem is solved using the Sequential Quadratic programming (DONLP) implementation in iSIGHT.

The BLISS decomposition consists of 4 subsystems including, Structures, Aerodynamics, Propulsion, and Range. A total of 6 system design variables are considered: thickness/chord ratio, altitude, Mach number, aspect ratio, wing sweep, and wing surface area. BB1 (Structures) has two local variables ($X1 = \text{taper ratio}$, wingbox cross-section), BB2 (Aerodynamics) has one local variable ($X2 = \text{skin friction coefficient}$), and BB3 (Propulsion) has one local design variable ($X3 = \text{throttle}$). BB4 computes the system objective Range and does not perform any local optimization.

The results obtained from BLISS/RS1 and BLISS/RS2 methods are compared with A-i-O and original BLISS methods in Table 3.2.4-1.

CASE	Initial Objective	Initial Max. Constraint Value	Final Objective	Final Max. Constraint Value	Computational Effort	
					Number of System Analyses	Number of Subsystem Analyses
A-i-O	535.79	-0.162	3964.19	+1.0e-08	119	(119x4x3) 1428
A-i-O/RS	535.79	-0.162	3974.84	+0.0013	72	(72x4x3) 864
BLISS	535.79	-0.162	3964.07	+1.92e-05	7 (6 BLISS cycles)	491
BLISS/RS1	535.79	-0.162	3961.5	+0.0	17 (4 BLISS cycles)	354
BLISS/RS2	535.79	-0.162	3964.12	+0.0	12 (11 BLISS cycles)	1097

Table 3.2.4-1: Aircraft Optimization results using BLISS/RS & comparison with A-I-O & BLISS

3.2.4.2 Conceptual Ship Design

MDO of a conceptual design of an oil tanker ship, where several disciplines are analyzed to provide one complete system analysis is considered. The disciplines involved in the system analysis include:

Hydrodynamics (BB1): involves engine propulsion calculations, wave and skin resistances (drag) modules, stability factor and range calculations;

Structures (BB2): involves weights and stress calculations; and,

Cost (BB3): total ship cost and the return-on-investment (ROI) computations.

The mathematical formulation of the A-i-O optimization problem is as follows:

Maximize: Return-on-Investment (ROI) = $f(X)$

Subject to:

Range = 10,000 Nm (+ 1.0%)

Displacement weight = 2×10^8 lbs (+ 1.0%)

Maximum (bending & shear) Stress < 30 ksi
 Stability factor < 0.0, and,
 Bounds on design variables.
 Six design variables, including, hull length, deck height, hull thickness, deck thickness, installed engine horse power, and fuel weight, are considered.

The BLISS decomposition consists of 3 subsystems (Hydrodynamics, Structures and Cost). The system level objective is to maximize ROI. A total of 3 system design variables are considered: Hull length, Deck height, and Fuel weight. BB1 has one local variable (X_1 = Installed HP) and local constraints on Range. BB2 has two local variables (X_2 = Hull thickness and deck thickness) and local constraints on displacement weight, bending and shear stresses. BB3 computes the ROI and does not perform any local optimization. All the local constraints and Stability requirement computed in BB1 are treated as system level constraints.

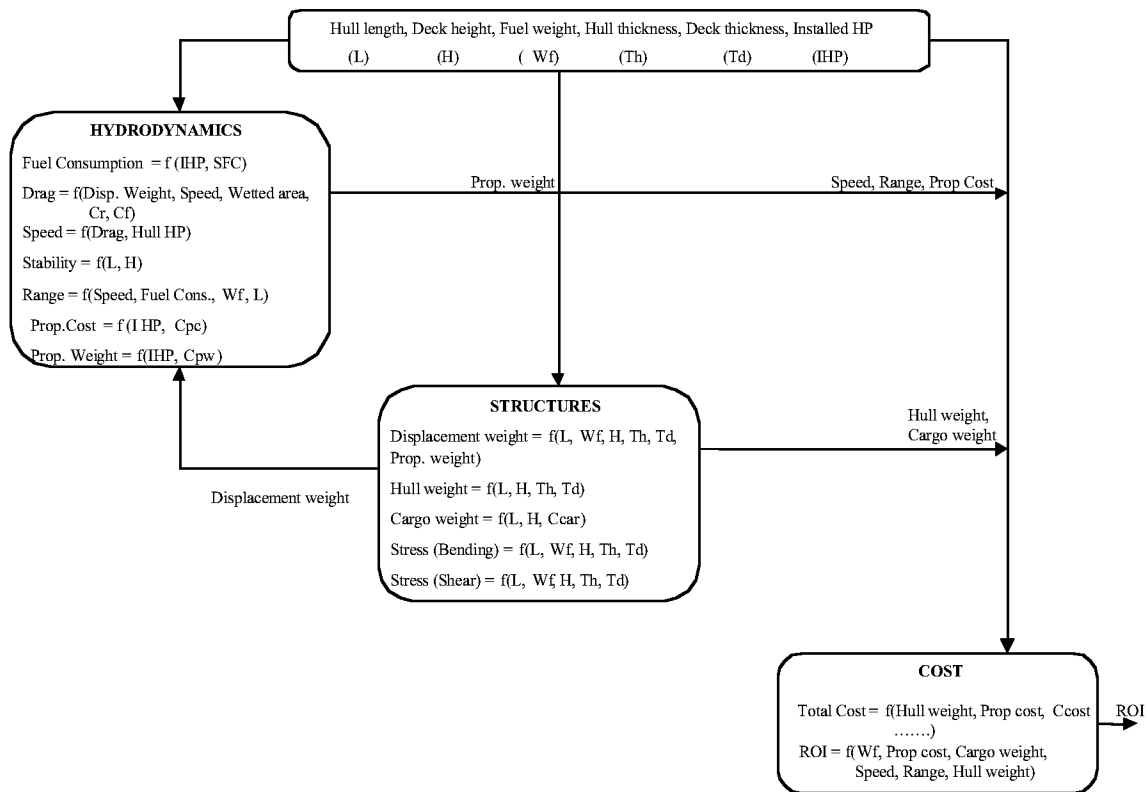


Figure 3.2.4-1: Conceptual ship analysis flow

Figure 3.2.4-1 shows a data flow diagram of one full system analysis for the problem. The results are provided in Table 3.2.4-2. The initial design is an infeasible design with an ROI of 0.2660. The ROI here represents (1/number of years to recover the

investment). In this example, the BLISS method did not arrive at the best known solution of 0.278 for the objective function (ROI).

CASE	Initial Objective	Initial Max. Constraint Value	Final Objective	Final Max. Constraint Value	Computational Effort	
					Number of System Analyses	Number of Subsystem Analyses
A-i-O	0.2660	+1.807	0.278	+0.003	111	(111x3x1) 333
A-i-O/RS	0.2660	+1.807	0.278	+0.002	50	(50x3x1) 150
BLISS	0.2660	+1.807	0.262	+0.002	46 (45 BLISS cycles)	5676
BLISS/RS1	0.2660	+1.807	0.266	+0.003	18 (14 BLISS cycles)	367
BLISS/RS2	0.2660	+1.807	0.270	+0.003	10 (9 BLISS cycles)	756

Table 3.2.4-2: Conceptual Ship Design results with BLISS/RS & comparison with BLISS & A-i-O procedures

3.2.5 Summary of BLISS/RS

The original BLISS (Sobieszczanski-Sobieski et al, 1998) method decomposes the problem into several optimizations at the component level that may be executed concurrently, and a coordinating optimization at the system level. With BLISS, the system-level were linked to the component-level optimizations by the optimum sensitivity derivatives. In BLISS/RS, the two optimization levels link through the Response Surfaces of a polynomial function type, and two variants of that linkage are introduced. In variant 1 the response surfaces for the system objective and the system constraints are constructed in the space of the system design variables using the system

analysis results. In variant 2, the response surfaces are being updated with the system component optimization results.

The method was demonstrated in application to a conceptual-level design of a supersonic business jet aircraft and a ship. Results were compared to those obtained by an all-in-one optimization (A-i-O), A-i-O with the response surfaces, and the original BLISS. In the aircraft test case, the method minimum objective agreed very well with that of the benchmark A-i-O. In the ship test case, the method fell short of the benchmark value by 4.3 %. In all the tests, the method showed a satisfactory capability to satisfy the constraints. In regard to the amount of numerical work, two different metrics were used. The metric equated to the number of the system analyses was found to be case-dependent. By that metric in the aircraft application, the method was not as efficient as BLISS but still an order of magnitude more efficient than A-i-O. In the ship case, the method was both an order of magnitude more efficient than A-i-O and about twice more efficient than the original BLISS. The other metric was the number of individual component analyses. Under that metric in the aircraft case, the method was more expensive than the original BLISS but still more economical than A-i-O, while in the ship application the method turned out to be more efficient than BLISS and about on par with A-i-O.

Interpreting the above results one should remember that the underlying analyses were exceedingly simple, typical of the conceptual design stage. One expects that the cost of the system analysis relative to the component analysis will increase as the design moved to the preliminary and detailed stages, hence the metric based on the number of the system analysis is likely to dominate.

Finally, the BLISS with Response Surfaces algorithm is well suited for exploiting the concurrent processing capabilities in a multiprocessor machine. Several steps, including the local sensitivity analysis, local optimization, response surfaces construction and updates are all ideally suited for concurrent processing. Needless to mention, such algorithms that can effectively exploit the concurrent processing capabilities of the compute servers will be a key requirement for solving large-scale industrial design problems.

3.3 Bi-Level Integrated System Synthesis for Structures (BLISS/S):

3.3.1 Introduction to BLISS/S:

The new method described herein is an adaptation of the method known as Bi-Level Integrated Synthesis (BLISS), Sobieski et. al., 1998, to applications in structures, hence the acronym BLISS/S (Sobieski and Kodiyalam, 1999). The key concept in the BLISS method was a decomposition of the design task into subtasks performed independently in each of the modules and a system-level or coordination task giving rise to a two-level optimization. In general, decomposition was motivated by the obvious need to distribute work over many people and computers to compress the task calendar time. Equally important benefit from the decomposition is granting an autonomy to the groups of engineers responsible for each particular subtask in choosing their methods and tools for the subtask execution. As an additional advantage, the concurrent execution of the subtasks fits well the technology of massively concurrent processing that is now becoming available. The above motivation and benefits apply also in large-scale structural optimization, especially for structures assembled of many dissimilar components or substructures. Applicability of two-level optimization to structures stems from the observation that, in general, a structure is defined by variables of two categories: the cross-sectional variables X , and the overall shape geometry variables Z . In optimization it is useful to distinguish between X and Z because:

- The X variables are associated with individual components and, therefore, they tend to be clustered. Also, the constraints they govern directly, e.g., the stringer buckling in built-up, thin-walled structures typical of aerospace vehicles, tend to be highly nonlinear. The total number of the X variables in a typical airframe is in thousands but their number in an individual substructure is likely to be quite small.
- The number of Z variables is much smaller than the total number of X variables.
- Nonlinearity of the overall behavior constraints, such as displacements, with respect to X and Z tends to be much weaker than that of the local strength constraints.
- Both Z and X influence entire structure, but the Z influence tends to be much stronger than that of X because it is exerted through the control of the structure overall shape while the X influence outside of the component they are associated with is governed by the degree of redundancy (that influence is zero in a statically determinate structure).

Accordingly, one may divide structural optimization procedure into two subtasks that alternate until convergence:

1. Separate, concurrently executed optimizations in the X -subspaces, each subspace corresponding to the X s associated with a component and dominated by the local, highly nonlinear constraints.
2. A single optimization in the Z -space in which only the displacement or frequency constraints of mild nonlinearity are present so that efficiency of Linear Programming may be exploited.

Because of the approximations involved at both levels, the optimizations in the X and Z spaces have to alternate iteratively until convergence.

The above decomposition is desirable not only for the reasons articulated in the foregoing but also because structural optimization performed in a conventional, all-in-one manner for a structure with a large number of design variables usually requires the use of approximations as surrogates of the full analysis in order to reduce the computational cost. However, if highly nonlinear constraints are present, the approximation error control requires imposition of narrow move limits in each stage based on the approximate analysis. That increases the number of stages required for convergence and may ultimately offset the intended benefit of the use of approximation as a cost control measure.

3.3.2 The BLISS/S Procedure

The BLISS/S procedure solves the optimization problem by decomposition. In optimization of i -th substructure performed for constant Z and Q, the local constraints depend on Z directly through the substructure geometry and, indirectly, through the influence of Z on Q. The X variables local to i -th substructure exert direct influence on the local constraints, and an indirect influence through the substructure stiffness coefficients that contribute to FEA and, therefore, to forces Q, not only the local ones but also to those acting on all substructures. Hence, the local X affects the optimization results in all substructures, not only the i^{th} substructure. The system-level displacement constraints are controlled by both Z and X that affect the substructure stiffness properties through its geometry, overall and cross-sectional. This web of influences is analogous to the one in a general modular system for which the original BLISS was developed.

Comparing to a general, multidisciplinary engineering system for which BLISS was originally developed, the structural system is degenerate in the sense that the inter-modular data exchange is limited to the flow of the internal forces data from the Finite Element Analysis Black Box to the Black Boxes representing the substructures (the structural elements). Also, the entire system analysis and sensitivity analysis are both contained within the Finite Element Analysis. Therefore, BLISS may be adapted to structures by substituting the elements of a structural optimization problem as follows.

A step-by-step prescription for BLISS/S is provided below for k^{th} cycle

1. (//) Update X and Z to the new values generated in the previous cycle $k-1$
(Initialize X and Z with the best guess if this is the first cycle).
2. (//) In BB_i for i^{th} substructure calculate the stiffness properties needed to represent the substructure in the FEA of the assembled structure. Repeat for all substructures.
3. Execute SA (FEA of the assembled structure) to compute the behavior variables: the displacement (u), the structure internal forces (Q) acting on the boundary of each substructure, and structural weight.
4. (//) Execute BBA for i^{th} substructure to compute its strength constraints (typically: stress and buckling), and its structural weight W_i . Repeat for all substructures. (optionally, the structural weight W_i for the substructures may be computed in Step 3 instead).

5. Check the termination criteria and continue or stop.
6. Execute SSA (FE Sensitivity Analysis) to obtain: $D(G^u, z_k)$, $D(W_i, z_k)$, $D(Q, z_k)$, $D(G^u, x_j)$, $D(Q, x_j)$, and for all elements in Z and X .
7. (//) Execute BBOSA for each BB_i to obtain $d(W_i, X_j)$, $d(g, Z_i)$ and $d(g, X_i)$, $d(g, Q_i)$

- 8.1 (//) If Z exist, execute BBOPT for BB_i ,

Find $_X$

Minimize $W = W^{k-1} + _j(d(W_i, X_j) + (D(W_i, Q_i)D(Q_i, X_j))) _X_j$

Satisfy $g_i \leq 0$; and $G^u \leq 0$

In the above approximate $G^u = (G^u)^{k-1} + D(G^u, X_i) _X$; Use $d(W_i, X_j)$ and $d(g, Q)$ from #7, $D(Q_i, X_j)$ from #6, and $D(W_i, Q_i)$ from #9 in the previous cycle $k-1$.

- 8.2 (//) If Z does not exist, execute BBOPT for BB_i ,

Find $_X$

Minimize $W = W^{k-1} + _j(d(W_i, X_j) + (D(W_i, Q_i)D(Q_i, X_j))) _X_j + _ d(G^u, X_j) _X_j$

Satisfy $g_i \leq 0$.

In the above approximate $G^u = (G^u)^{k-1} + D(G^u, X_i) _X$; Use $d(W_i, X_j)$, and $d(g, Q)$ from #7; $D(G^u, X_i)$ and $D(Q_i, X_j)$ from #6; and $D(W_i, Q_i)$ from #9 in the previous cycle $k-1$.

9. (//) Execute BBOSA to compute $D(W_i, Z_k)$ and $D(W_i, Q_i)$ using (as in #7, BLISS/B) the algorithm from Barthelemy and Sobieszczanski-Sobieski, 1983. The algorithm is summarized in Appendix.

10. Execute SOPT, IF Z present, BYPASS if Z absent

Find $_Z$

Minimize $W = _W_i$

Satisfy $G^u \leq 0$

In the above, approximate

$W_i = (W_i)_o + (D(W_i, Z) + (D(W_i, Q_i) D(Q_i, Z))) _Z$ using $D(W_i, Z)$, $D(W_i, Q_i)$ from BBOSA in #9, and $D(Q, Z)$ from SSA in #6

$G^u = (G^u)_o + D(G^u, Z) _Z$ using the derivatives from SSA in #6.

11. Begin the next cycle, $k=k+1$, from #1.

Notes:

- 1) In the first cycle, $k=1$, set $D(W_i, z_k) = 0$ and $D(W_i, Q_i) = 0$;

- 2) Concurrent, coarse-grained, processing opportunities are marked by (/). They are the opportunities created by BLISS/S. The other opportunities that might be intrinsic in FEA and FE Sensitivity Analysis, e.g., concurrent processing of many right-hand side vectors in structural sensitivity analysis are not marked.
- 3) In #8.2, BBOPT is a sole means of satisfying G^u , hence the penalty term appended to the objective. The penalty factor $_$ should be set so that, initially, the penalty term magnitude is of the same order as the other terms in the expression.
- 4) In #8.1 and #8.2, any suitable search algorithm may be used. It does not have to be the same for all substructures.
- 5) In #10, owing to the linearization of all the functions, one may use a Linear Programming technique to obtain $_Z$

As mentioned before, the key to effectiveness of the BLISS/S procedure is a judicious use of the sensitivity information. In BBOPT, step #8, the direct influence of X_i on W_i is captured by $d(W_i, X)$ and the indirect influence of X through the change of Q due to the change of X is represented by the term $D(W_i, Q_i) D(Q_i, X_j)$. As mentioned before, this indirect influence is important in redundant structures, and in contrast to the term $d(W_i, X)$ that is purely local, the term reflects the influence of X_i on the entire structure.. Similarly, in SOPT, the term $(D(W_i, Z) + (D(W_i, Q_i) D(Q_i, Z)))$ plays an analogous role for W_i and Z .

Although SOPT, step #10 does not explicitly address the local constraints g , satisfaction of these is protected by the use of $D(W_i, Z)$, and $D(W_i, Q_i)$ obtained from BBOSA, step #9. It is so because the algorithm of Barthelemy and Sobieszczanski-Sobieski, 1983 (see also Appendix) generates these derivatives as constrained derivatives. In other words the algorithm treats Z and Q as the parameters of the optimization that was executed in BBOPT, and coordinates the changes $_W_i$, $_Z$, and $_Q_i$ so as to preserve $g = 0$.

Finally, one should note that when there are no variables Z , the procedure becomes a special case of optimization by piece-wise linear approximations as SOPT in Step #10 is bypassed. However, even in this case the optimization remains decomposed because each substructure is optimized separately in Step #8.

3.3.3 BLISS/S Procedure with Response Surface Approximations for Substructure Analysis:

As emphasized in the foregoing, the substructure optimizations are independent of each other and unrestricted in regard to the choice of the method. The use of a Response Surface (RS) Method to represent the BBA operation is an example.

In this application a polynomial response surface-based optimization employs an “adaptable” response surface model (RSM) in place of the substructure analysis. That model is implemented in iSIGHT software (Golovidov et al., 1998). In this approach, a minimum number of designs are used to construct an initial RS around the baseline design. Typically, a linear RS is constructed initially, although the user has an option to request a quadratic initial RS. For a linear RS, this number would be $(N_{inp}+1)$, where N_{inp} is the number of inputs. After the best design is found using this RS within the specified design space bounds, the design is analyzed using the "exact analysis", the data are included into the RS data set, and the RS is regenerated.

3.3.4. BLISS/S Numerical Examples and Results

The validation results include test cases compared to the benchmark all-in-one (A-i-O) optimization for accuracy of the final results and the convergence characteristics. The validation test case is a hub framework that appears in Balling and Sobieszczanski-Sobieski, 1994. Utility of the hub structure as an optimization test case stems from its ability to include as many members as desired without increasing the dimensionality of the load-deflection equations. These equations remain 3×3 for a 2D hub structure regardless of the number of members. While analytically simple, the hub structure design space is complex because the stress, displacement, and buckling constraints are rich in nonlinearities and couplings among the design variables.

In the hub structure herein, each beam has an I-shaped cross-section. The X variables are the dimensions of b_1 , h , b_2 , t_1 , t_2 , and t_3 . The top and bottom flanges of the I-beam are not of the same dimensions, hence the cross-section of each I-beam requires 6 design variables. The Z variables are the horizontal and vertical coordinates of the hub point where the beam members are rigidly connected. The change in the coordinates of the hub results in the change of the angles between the beams.

An example of g is the local buckling of the top flange in beam #2, and an example of G is the horizontal displacement of the hub. The constraint formulation details may be found in Balling, and Sobieszczanski-Sobieski, 1994.

The BLISS/S procedure was tested on the above structure in a 2-beam, 4-beam and 60-beam versions, all considered under one or two of the loading conditions specified in the appendix. The three versions had, respectively, (12 local design variables, 76 local constraints), (24 local design variables, 76 local constraints), and (360 local design variables, 1140 local constraints). The local constraints included the stress and local buckling constraints for both loading conditions. In both versions, the system-level constraints were imposed on the resultant translations and one rotation at the hub point for each of the two loading conditions. System design variables, corresponding to the hub location are also considered.

The tests were organized in three cases beginning with the one in which Z and G are absent but X and g present, and ending with Z, G, X, and g all present. Each table is labeled with the case description and shows the objective function (structural weight), and the maximum constraint values for the initial and optimal states for the benchmark A-i-O method, BLISS/S, and BLISS/S/RS.

The benchmark A-i-O method is a piece-wise approximate optimization in which there is no decomposition. The structure FEA includes computation of gradients by finite differences (one-step-forward) that are then used to form a linear extrapolation as an approximate analysis for optimization within move limits. The optimizations within move limits were performed by the usable-feasible directions method. The same method was used at the substructure and system levels in BLISS/S. The gradients at both levels were also computed by finite differences for consistency of comparison with the A-i-O method, except the derivatives of the objective with respect to the optimization parameters that were computed by the algorithm described in the appendix. The BLISS/S procedure was implemented in the software framework called iSIGHT (1998).

As a measure of the numerical labor, the tables, show how many calls were issued to the assembled structure FEA and the total of such calls to the substructure analyses. The latter is not shown for the benchmark method. In that method the substructure analyses are a part of SA because there is no decomposition so that each substructure is analyzed once in each execution of SA.

3.3.4.1 Example 1 – 2-Beam Model, Case 1, 2 and 3:

Tables 3.3.4-1 to 3.3.4-5 and Figures 3.3.4-2 and 3.3.4-3 document the results of this example problem for all 3 cases. The tables show that the difference between the BLISS/S objective minimum and the benchmark remains well under 1 %, except for BLISS/S/RS where it reaches 2.7 % for the two-member case. Regarding the comparison of the individual design variables, the volume of data is too large to show in full, therefore, only a typical sample is given in Table 3.3.4-2.

Examination of the data showed that, as expected, the discrepancies for the individual design variables are greater than those for the objective function. In terms of the numerical labor, BLISS/S shows significant reduction of the number of calls to the assembled structure analysis. To be fair one should note that the BLISS/S advantage in this regard is amplified by the use of finite difference gradients in both BLISS/S and in the benchmark method. That advantage would be less if analytical gradient calculation was used in both methods. One should emphasize that one potential advantage of BLISS not tested in this report is its amenability to concurrent execution of the substructure optimizations and associated analyses.

The case in Table 3.3.4-4 comprises two subcases labeled 10 % and 100 %. The percentage labels refer to the side constraints imposed on the horizontal and vertical location of the hub. For example, in the 10 % subcase the horizontal coordinate of the hub location, a Z variable, was restricted to be less or equal to 0.1 length of the horizontal member. Thus, the 100 % subcase is special so that it allows an extreme reconfiguration of the structure such that the hub moves all the way to the root of the horizontal member. The expected result was that given that freedom, the procedure should eliminate the one of the members so that the load would be applied directly to the wall. The remaining member should then shrink to minimum gages resulting in a very light, degenerate structure. The results in Table 3.3.4-4 confirmed the above expectation, and Fig. 3.3.4-1 illustrates the corresponding reconfiguration of the structure.

Histogram in Fig. 3.3.4-2 shows the objective function convergence for the benchmark method and BLISS/S. It indicates that BLISS/S converges the objective to the benchmark value within 5 cycles, effectively recovering from the error in extrapolation based on the optimum sensitivity derivatives. A similar phenomenon was reported in Sobieszczanski-Sobieski, 1993. The histogram in Fig. 3.3.4-3 shows the most violated constraint convergence. In this regard, BLISS/S holds the most violated constraint satisfied from start to finish. In contrast, the benchmark method allows one of the initially feasible constraints to become violated and displays an irregular, oscillatory convergence of that constraint. This confirms the expectation that decomposition in BLISS/S makes satisfaction of highly nonlinear local constraints easier.

Table 3.3.4-1: Two Beam Hub Frame Solutions for Case 1

Case	Initial Design Objective	Initial Max Constraint Violation	Final Design Objective	Final Max Constraint Violation	Number Of System FEA	Number of Substructure Analyses
A-I-O	1988.0	-0.162662	1045.5	0.00093	165 10 cycles	
BLISS/S	1988.0	-0.162662	1045.5	0.00132	131 10 cycles	841
BLISS/S (with RSA in BBOPTj)	1988.0	-0.162662	1073.45	-0.0084	79 6 cycles	486

Table 3.3.4-2: Comparison of design variable values for Case 1

Design Variable	Initial Value (cm)	A-I-O Final Value (cm)	BLISS/S Final (cm)	BLISS/S Final (w/ RSA)
M1-b1	5.0	4.47	5.04	4.66
M1-b2	5.0	4.43	4.44	4.39
M1-b3	0.4	0.26	0.21	0.12
M1-t1	0.4	0.28	0.27	0.34
M1-t2	0.4	0.27	0.26	0.22
M1-h	5.0	2.73	2.81	4.49
M2-b1	5.0	5.86	6.0	6.0
M2-b2	5.0	5.86	5.99	5.44
M2-b3	0.4	0.17	0.19	0.17
M2-t1	0.4	0.21	0.20	0.24
M2-t2	0.4	0.21	0.19	0.20
M2-h	5.0	3.49	3.69	3.46

Note: M1 and M2 are the member numbers

Table 3.3.4-3: Two Member Hub Frame Solutions for Case 2

Case	Initial Design Objective	Initial Max Constraint Violation	Final Design Objective	Final Max Constraint Violation	Number Of System FEA	Number of Substructure Analyses
A-I-O	1988.0	-0.162662	1538.16	0.00082	106 6 cycles	
BLISS/S	1988.0	-0.162662	1537.48	-0.0065	79 6 cycles	503
BLISS/S (with RSA in BBOPTj)	1988.0	-0.162662	1592.83	0.0022	79 6 cycles	268

Table 3.3.4-4: Two Member Hub Frame Solutions for Case 3

Case	Initial Design Objective	Initial Max Constraint Violation	Final Design Objective	Final Max Constraint Violation	Number Of System FEA	Number of Substructure Analyses
10% A-I-O	1988.0	-0.162	1447.97	0.0039	242 14 cycles	
10% BLISS/S	1988.0	-0.162	1470.89	0.0037	211 14 cycles	1295
100% A-I-O	1988.0	-0.162	8.55796	-0.140	497 30 cycles	
100% BLISS/S	1988.0	-0.162	8.33236	-0.0031	226 15 cycles	2219

Table 3.3.4-5: Two member hub frame – Location variable comparison for Case 3 (100%)

Hub Location	Initial Location (cm)	A-i-O Final Location	BLISS/S Final
X	150.0	0.93	1.05
Y	20.0	19.9	19.8

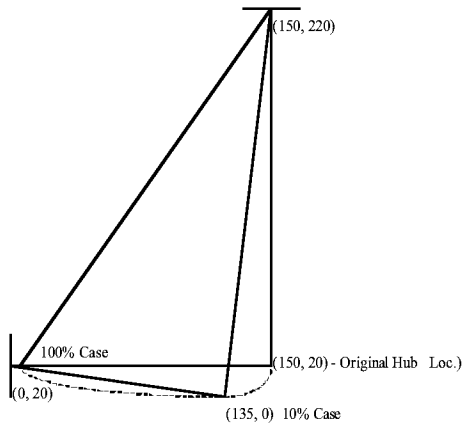


Figure 3.3.4-1. Hub structure, 2-Beam Model: optimal hub location for 10 % and 20 % side constraint.

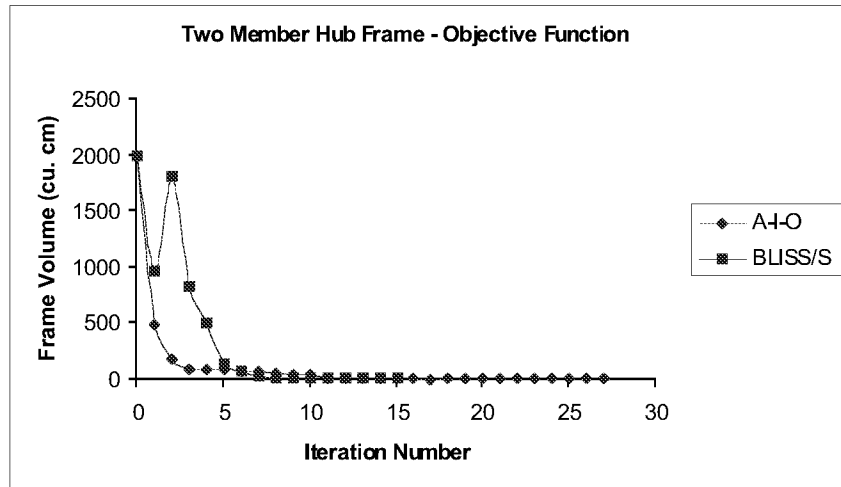


Figure 3.3.4-2. Histogram of the 2-beam, hub frame model objective function (Case 3, 100%).

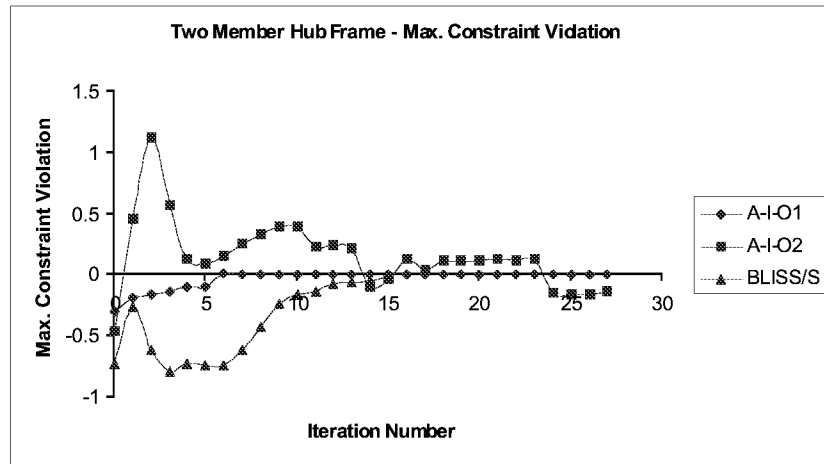


Figure 3.3.4-3. Histogram of the 2-beam, hub frame model maximum constraint violations (Case 3, 100%).

3.3.4.2 Example 2 – 4-Beam Model, Case 3:

BLISS/S is used to solve 4-beam hub frame model for Case 3, shown in Figure 3.3.4-4. A single loading condition at the hub location with a value of 300 kN is considered. One system design variable, Z_v , is defined, which is the vertical displacement. There are 2 system constraints, translational and rotational displacement of the frame at the hub point. Each of the 4 members has 6 design variables and 19 stress and buckling constraints. Overall, there are 24 local design variables and 76 local constraints, 1 system design variable and 2 system constraints. For BLISS/S, each member is treated as a sub-system. As a result, there are 4 sub-systems and each sub-system has six design variables – section variables, $\{b_1, b_2, b_3, t_1, t_2, h\}$. The initial position for the hub point is off centered, ($Z_v=10$). The final solution with the hub point moving to the center is shown in Figure 3.3.4-4. Both A-i-O and BLISS/S solutions show that member 1, which is at 9 o'clock position, is in tension with some of its section variables reaching their upper bounds. The results are summarized in Tables 3.3.4-6 through 3.3.4-8 and the iteration histories are shown in Figures 3.3.4-5 and 3.3.4-6.

Table 3.3.4-6: 4 Member Hub Frame Case 3

Case	Initial Design Objective	Initial Max Constraint Violation	Final Design Objective	Final Max Constraint Violation	Number of System FEA	Number of Substructure Analyses
A-i-O	3979.78	0.0644	2031.97	-2.38E-07	426	
BLISS/S	3979.78	0.0644	2030.53	0.00228	5 cycles 131	2277

Table 3.3.4-7: 4 Member Hub Frame Case 3 – 9 o'clock member section variables

	Initial	Final
A-i-O	{5.0, 5.0, 0.4, 0.4, 0.4, 5.0}	{5.1, 6.0 , 0.42, 0.12, 0.27, 6.47}
BLISS/S	{5.0, 5.0, 0.4, 0.4, 0.4, 5.0}	{ 6.0 , 5.18, 0.45, 0.35, 0.31, 5.98}

Table 3.3.4-8: 4 Member Hub Frame System Variables

	Initial Value	Final Value	Lower Bound	Upper Bound
Zv	10.0	19.4427	0.0	40.0

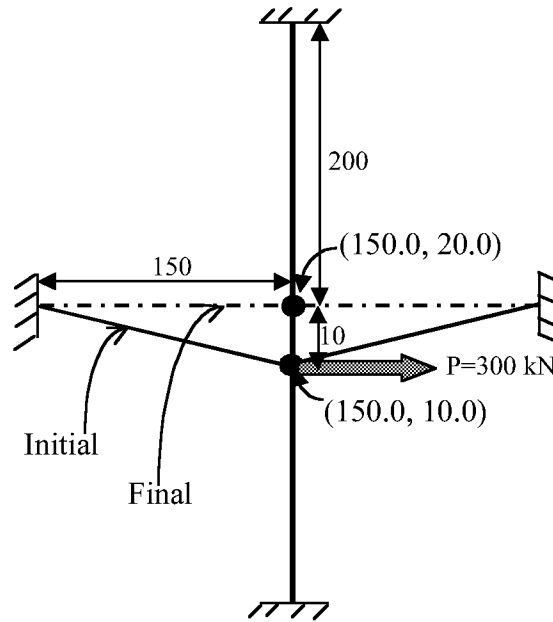


Figure 3.3.4-4: 4-beam, Hub Frame model: initial and final configurations

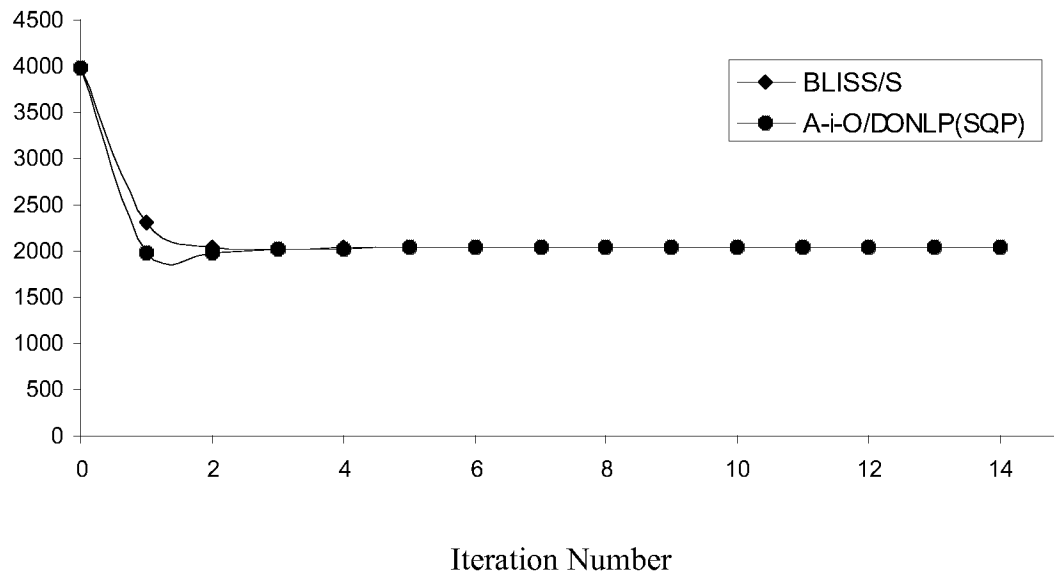


Figure 3.3.4-5: Histogram of 4-beam, Hub frame problem objective function (Case 3)

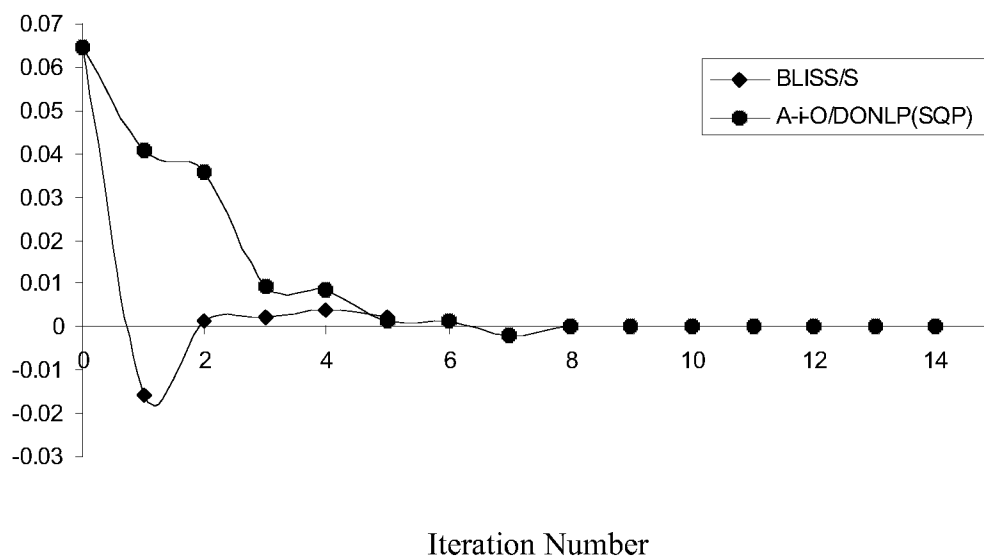


Figure 3.3.4-6: Histogram of 4-beam, Hub frame problem maximum constraint violation (Case 3)

3.3.4.3 Example 3 – 60-Beam Model, Case 1 and 3:

BLISS/S is used to solve 60-beam hub frame problem for Cases 1 and 3. A single loading condition at the hub with a value of 600 kN is considered. The sixty members are 6 degrees apart with each other. Each of the 60 members has 6 design variables and 19 stress and buckling constraints. Overall, there are 360 local design variables and 1140 local constraints. In Case 1, there are 60 sub-systems and each sub-system has six design variables – section variables, $\{b_1, b_2, b_3, t_1, t_2, h\}$. The 9 o'clock member is only half as long as all the other members. This makes it potentially stiffer and predisposed to attract the internal forces to transmit the hub load to the wall at the least expense in structural weight. BLISS/S confirms this expectation. It yields a better solution than A-i-O methods and its solution has the 9 o'clock member in tension at the final design. Case 1 results are provided in Tables 3.3.4-9 and 3.3.4-10. Table 3.3.4-9 shows that the BLISS/S solution has a final volume of 8585.3 cu.cm, 50% lower than the final volume obtained by A-i-O/Conmin and nearly 14% lower than the final volume obtained by A-i-O/SLP.

For case 3, in addition to the local variables, one system design variable, Z_v is considered. There are 2 system constraints corresponding to the translational and rotational displacement of the frame. To create a situation similar to that described for Example 2, the initial position of the hub point is off centered ($Z_v=10$). In this problem, BLISS/S yields a much better solution than A-i-O approach. As expected, in the BLISS/S solution, the hub has moved to the center ($Z_v=20.1267$, Table 13) and most of the load is efficiently transmitted through tension of the 9 o'clock member. Some of that member cross-sectional dimensions have grown to upper bounds at the expense of shrinkage in other members. The A-i-O approach failed to converge to this solution. Case 3 results are provided in Tables 3.3.4-11, 3.3.4-12, and 3.3.4-13. As Table 3.3.4-11 indicates, the BLISS/S solution has a final volume of 8558.8 cu.cm and this is 13% lower than the final volume obtained by A-i-O/MMFD and 19% lower than the final volume obtained by A-i-O/SLP.

Table 3.3.4-9: 60-Member, Hub Frame model, Case 1

	Initial Design Objective	Initial Max Constraint Violation	Final Design Objective	Final Max Constraint Violation	Number of System FEA	Number of Substructure Analysis
A-i-O/Conmin	65431.0	-0.688	17253.0	0.000025	3664	
A-i-O/SLP	65431.0	-0.688	9948.79	0.002377	5423	
BLISS/S	65431.0	-0.688	8585.36	0.001	20 cycles 7221	55669

Table 3.3.4-10: 60-Member, Hub Frame, Case 1 – 9 o'clock member section variables

Case	Initial	Final
A-i-O/Conmin	{5.0, 5.0, 0.4, 0.4, 0.4, 5.0}	{5.00, 5.00, 0.389, 0.386, 0.386, 5.00}
A-i-O/SLP	{5.0, 5.0, 0.4, 0.4, 0.4, 5.0}	{5.20, 5.20, 0.945, 0.917, 0.917, 5.19}
BLISS/S	{5.0, 5.0, 0.4, 0.4, 0.4, 5.0}	{ 6.0 , 6.0 , 0.972, 1.0 , 0.579, 7.79}

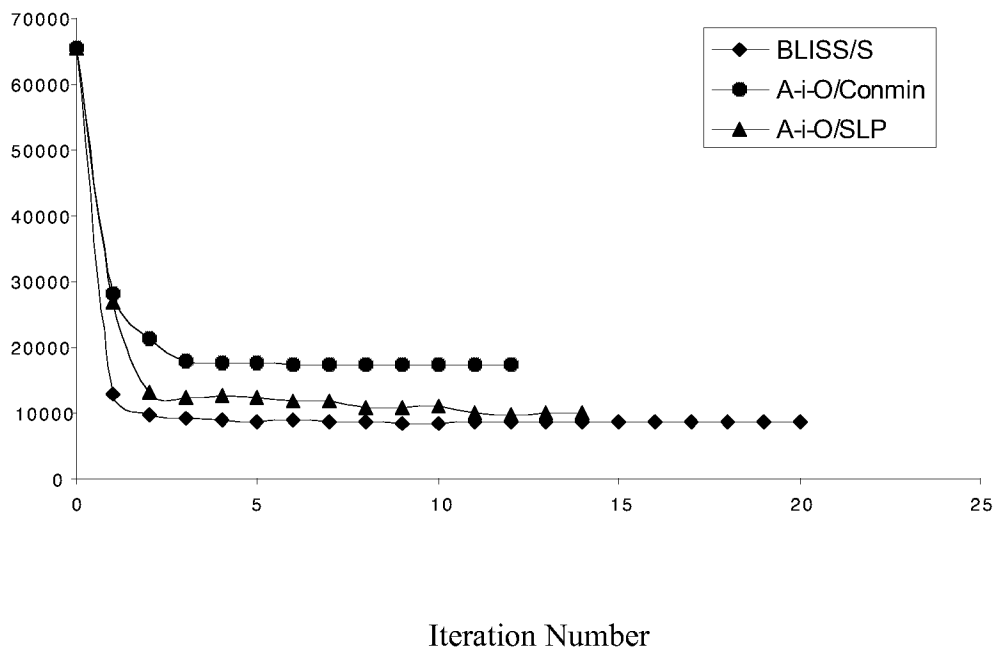


Figure 3.3.4-7: 60-member, Hub frame model, Objective function iteration history, Case 1

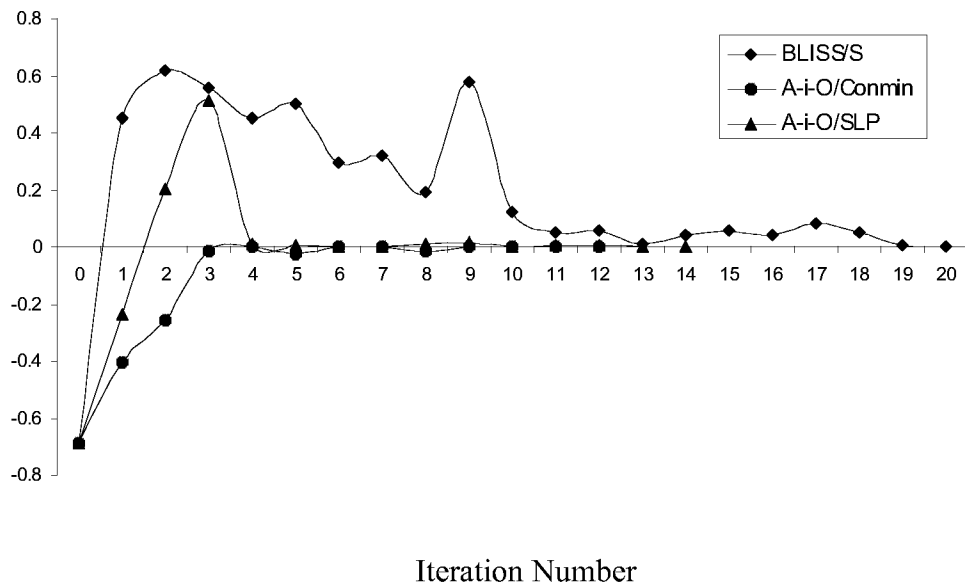


Figure 3.3.4-8: 60-member, Hub frame model, max. constraint violation iteration history, Case 1

Table 3.3.4-11: 60 Member Hub Frame Case 3

Case	Initial Design Objective	Initial Max Constraint Violation	Final Design Objective	Final Max Constraint Violation	Number of System FEA	Number of Substructure Analyses
A-i-O/MMFD	65843.0	-0.685	12716.1	0.00224	5503	
A-i-O/SLP	65843.0	-0.685	10564.0	0.00299	5800	
BLISS/S	65843.0	-0.685	8558.81	0.00296	11 cycles 3983	31991

Table 3.3.4-12: 60 Member Hub Frame Case 3; 9 o'clock member section variables

Case	Initial	Final
A-i-O/MMFD	{5.0, 5.0, 0.4, 0.4, 0.4, 5.0}	{5.05, 5.05, 0.47, 0.807, 0.98, 4.41}
MDF-SLP	{5.0, 5.0, 0.4, 0.4, 0.4, 5.0}	{4.91, 5.30, 1.0 , 0.996, 1.0 , 4.41}
BLISS/S	{5.0, 5.0, 0.4, 0.4, 0.4, 5.0}	{ 6.0 , 6.0 , 0.75, 1.0 , 1.0 , 6.70}

Table 3.3.4-13: 60 Member Hub Frame Case 3; System Variables

	Initial Value	Final Value	Lower Bound	Upper Bound
Z _v	10.0	20.1267	0.0	40.0

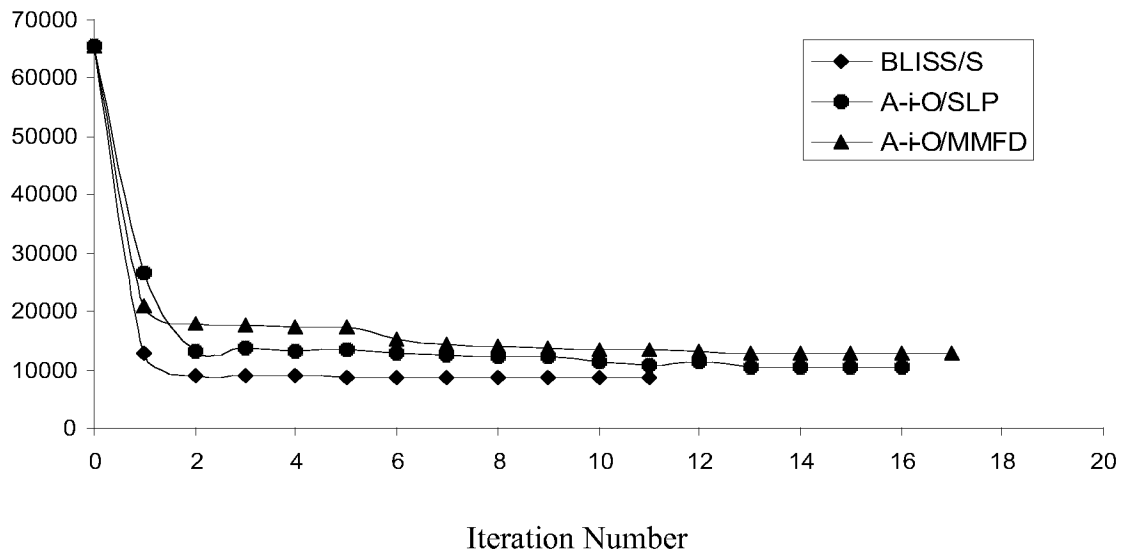


Figure 3.3.4-9: 60-member, Hub frame model, objective function iteration history, Case 3.

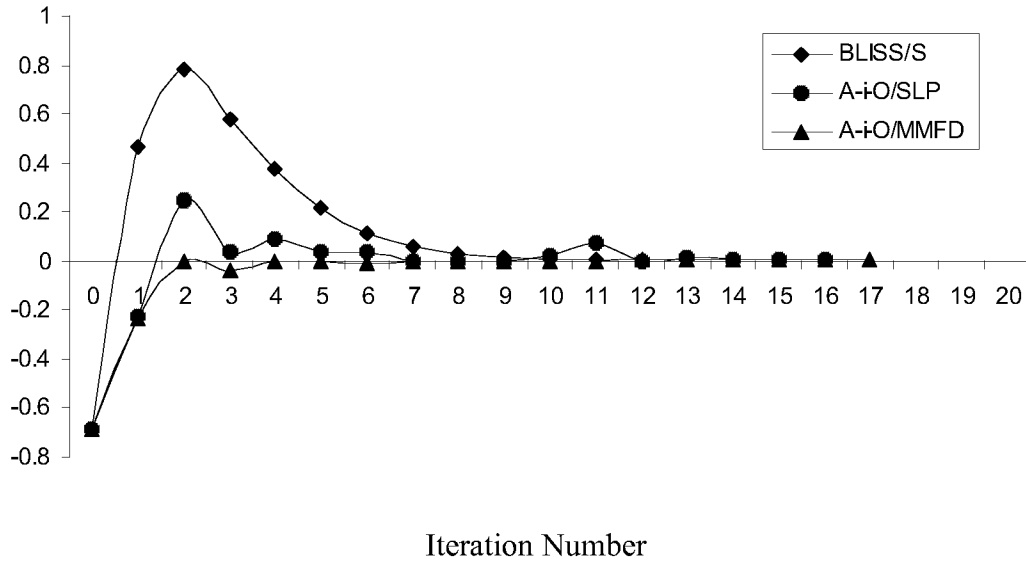


Figure 3.3.4-10: 60-member, Hub frame model, Max. constraint violation iteration history (Case 3)

3.3.5 BLISS/S Extensions – Active constraints switching & Computation of Lagrange multipliers of active constraints external to the numerical optimizer:

As mentioned earlier, the key to effectiveness of the BLISS/S procedure is a judicious use of the sensitivity information. Although the system optimization (SOPT), step #10 does not explicitly address the local constraints g , satisfaction of these is protected by the use of $D(W_i, Z)$, and $D(W_i, Q_i)$ obtained from the black box optimal sensitivity analysis (BBOSA), step #9. It is so because the algorithm of Barthelemy and Sobieszczanski-Sobieski, 1983 generates these derivatives as constrained derivatives. In other words the algorithm treats Z and Q as the parameters of the optimization that was executed in BBOPT, and coordinates the changes $_W$, $_Z$, and $_Q$ so as to preserve $g=0$.

It is important to note here the BBOSA calculations for $D(W_i, Z)$, and $D(W_i, Q_i)$ involve the use of lagrange multipliers of the active constraints of the BBOPT solution. Any switching of the active local constraints between successive BLISS/S cycles could therefore results in oscillation and delay of convergence of the overall BLISS/S process. In order to investigate this active constraint switching and its influence on the overall convergence, the Lagrange (Kuhn-Tucker) multipliers of the currently active and previously active constraints of the i^{th} BBOPT problem are computed outside of the numerical optimizer. This computation involves the solution of

$$[GC] \{L\} = [GF]$$

where $[GC]$ is the matrix of the active constraint derivatives, $[GF]$ is the matrix of objective function derivatives and $\{L\}$ is the vector of unknowns (Lagrange multipliers).

This modification to the BLISS/S procedure was tested on the 60 member hub frame model (both Case 1 and Case 3). The results are documented in Tables 3.3.5-1 and 3.3.5-2.

Table 3.3.5-1: 60-Member Hub Frame Case 1 – BLISS/S with extensions

	Initial Design Objective	Initial Max Constraint Violation	Final Design Objective	Final Max Constraint Violation	Number of System FEA	Number of Substructure Analysis
BLISS/S	65431.0	-0.688	8585.36	0.001	20 cycles 7221	55669
BLISS/S with Extensions	65431.0	-0.688				

Table 3.3.5-2: 60 Member Hub Frame Case 3 – BLISS/S with Extensions

Case	Initial Design Objective	Initial Max Constraint Violation	Final Design Objective	Final Max Constraint Violation	Number of System FEA	Number of Substructure Analyses
BLISS/S	65843.0	-0.685	8558.81	0.00291	11 cycles 3983	31991
BLISS/S with Extensions	65843.0	-0.685	8573.63	0.0039	11 cycles 3983	44407

From the Tables 3.3.5-1 and 3.3.5-2, it is reasonable to conclude that the switching of active constraints (local), if any, does not slow down the convergence of the BLISS/S process. However, with the modified procedure (BLISS/S with Extensions) the maximum violated constraint does not switch between the successive BLISS/S cycles as much compared to the original BLISS/S procedure.

3.3.6 Summary of BLISS/S:

A two-level optimization method known from previous publication as BLISS for Bi-Level System Synthesis was adapted to structural optimization purposes and labeled BLISS/S for BLISS/Structures. The original method decomposes a modular system optimization into subtask optimization, that may be executed concurrently, and the system optimization that coordinates the former. Transformation of BLISS into BLISS/S was accomplished by treating the substructures (ultimately, the individual members) as modules in a generic system and by specifying the Finite Element Analysis as the equivalent of the system analysis. The resulting procedure separates the multitude of the cross-sectional variables from the overall structure geometry (shape) variables. Also, the highly non-linear local constraints, e.g., the local buckling, remain in the individual

substructure optimizations and do not directly enter the assembled structure optimization performed under the system constraints, e.g., the displacement constraints.

The validation tests performed using a hub framework structure with up to 60 members (361 variables, 1142 constraints) showed satisfactory agreement with the benchmark results obtained by optimization without decomposition, in terms of the minimum of the objective. They showed that the BLISS/S ability to satisfy the local constraints is better than that of the benchmark method. This advantage is expected to be amplified with the increase of the number of the substructures and the degree of non-linearity of the constraints. In terms of the numerical labor, the results showed that BLISS/S reduces that labor substantially with regards to the number of full FEA but requires additional substructure analyses for the substructure optimizations. Therefore, the degree of the pay-off from the use of BLISS/S instead of the A-i-O method will increase with the number of the design variables if the problem is large enough so that the cost of the FEA is dominant. In some of the particular cases tested, the BLISS/S solutions for the final design objective values were significantly better than those obtained by the benchmark optimizations that used no decomposition. In addition, the results showed the BLISS/S ability to satisfy the local constraints is better than that of the benchmark methods.

The BLISS/S additional advantage is its amenability to execute substructure optimizations concurrently and autonomously so that different optimization techniques may be used if these substructures are heterogeneous. Implementation of BLISS/S in a heterogeneous computing environment to exploit the latter advantage is the future development of a potentially high pay-off.

3.4 Automotive MDO Application using Massively Parallel Processing:

3.4.1 Introduction to MDO Application Problem:

To be competitive on the today's market, cars have to be as light as possible while meeting the Noise, Vibration, and Harshness (NVH) requirements and conforming to Government-mandated crash survival regulations. The latter are difficult to meet because they involve very compute-intensive, non-linear analysis, e.g., the code RADIOSS capable of simulation of the dynamics, and the geometrical and material nonlinearities of a thin-walled car structure in crash, would require over 12 days of elapsed time for a single design of a 390K elastic degrees of freedom model, if executed on a single processor of the state-of-the-art SGI Origin2000 computer. Of course, in optimization that crash analysis would have to be invoked many times. Needless to say, that has rendered such optimization intractable until now. The car finite element model is shown in Figure 3.4.1-1.

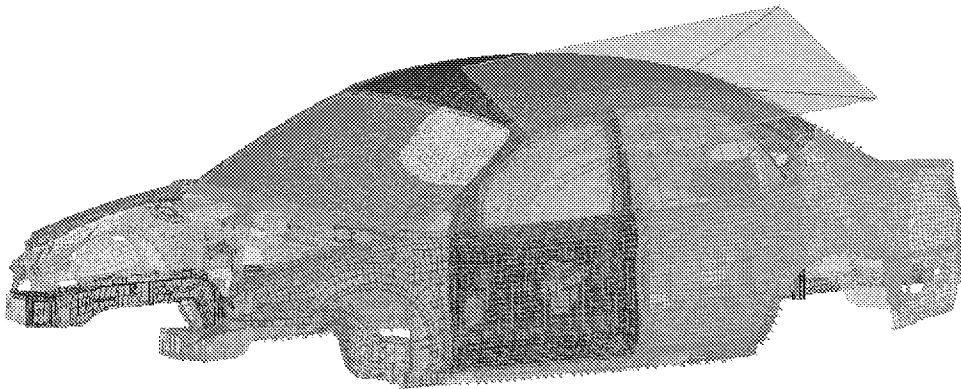


Figure 3.4.1-1 Car Body Finite Element Model

Some details of the model include:

```
NUMMAT: NUMBER OF MATERIALS ..... 462
NUMNOD: NUMBER OF NODAL POINTS ..... 128826
NUMBCS: NUMBER OF BOUNDARY CONDITIONS ..... 6085
N2D3D : ANALYSIS TYPE: 0=3D,1=AXISYM,2=PLANE STRAIN . 0
NUMELQ: NUMBER OF 2D SOLID ELEMENTS ..... 0
NUMELS: NUMBER OF 3D SOLID ELEMENTS ..... 0
NUMELC: NUMBER OF 3D SHELL ELEMENTS (4-NODES) .... 124868
NUMELT: NUMBER OF 3D TRUSS ELEMENTS ..... 0
NUMGEO: NUMBER OF PROPERTY SETS ..... 286
NUMELP: NUMBER OF 3D BEAM ELEMENTS ..... 2
NUMELR: NUMBER OF 3D SPRING ELEMENTS ..... 2484
NUMELTG: NUMBER OF 3D SHELL ELEMENTS (3-NODES) ... 0
```

3.4.2 MDO Problem Statement:

3.4.3 Solution Procedure:

The advent of computers that comprise large numbers of concurrently operating processors has created a new environment wherein the above optimization, and other engineering problems heretofore regarded as intractable may be solved. A flow chart of solution procedure is shown in Figure 3.4.3-1. The procedure is a piecewise approximation based method and involves using a sensitivity based Taylor series approximation model for NVH and a polynomial response surface model for Crash. In that method the NVH constraints are evaluated using a finite element code (MSC-NASTRAN) that yields the constraint values and their derivatives with respect to design variables. The crash constraints are evaluated using the code RADIOSS (from Mecalog) on the Origin 2000 operating on 256 processors simultaneously to generate data for a polynomial response surface in the design variable domain. The NVH constraints and their derivatives combined with the response surface for the crash constraints form an approximation to the system analysis (surrogate analysis) that enables a cycle of multidisciplinary optimization within move limits. In the inner loop, the NVH sensitivities are recomputed to update the NVH approximation model while keeping the Crash response surface constant. In every outer loop, the Crash response surface approximation is updated, including a gradual increase in the order of the response surface and the response surface extension in the direction of the search.

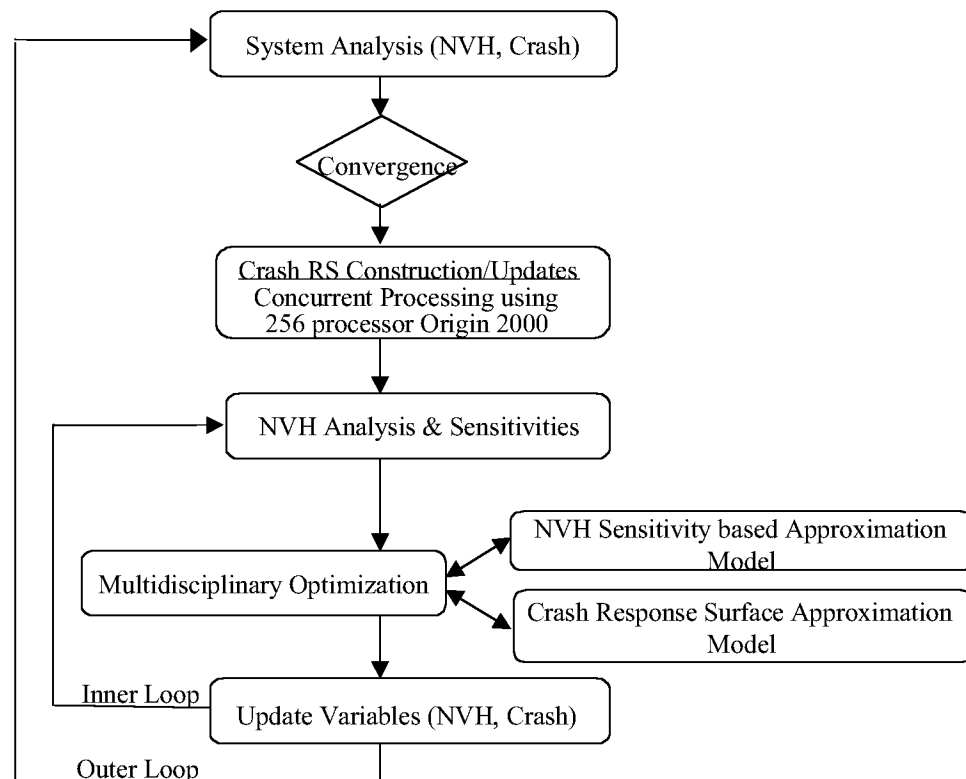


Figure 3.4.3-1 Flow chart of the solution procedure

In this optimization task, the NVH discipline has 30 design variables while the crash discipline has 20 design variables. A subset of these design variables (10) are common to both the NVH and crash disciplines. The number of design points used for the constructing the initial (linear) response surface for the Crash constraints were 25 design points. A design of experiment procedure was used to generate a 2 level design matrix of 24 design points and the 25th design point corresponds to the baseline design. On a single processor in Origin 2000 that amount of computing would require over 10 months! As mentioned previously, these runs were carried out concurrently on the Origin 2000 using multiple processors, ranging from 8 to 16, for each crash (RADIOSS) analysis.

Figure 3.4.3-2 shows the wall time required for a single RADIOSS analysis using varying number of processors. Figure 3.4.3-2 also shows a comparison of 2 different common data placement procedures within the allotted memories for each analysis. With the Origin 2000 computer, each memory is associated with 2 CPUs by default. When an analysis solution is performed, for example, using 12 CPUs, 6 different local memories are involved. The common data placement procedures that are compared here include, ROUND ROBIN and FIRST TOUCH.

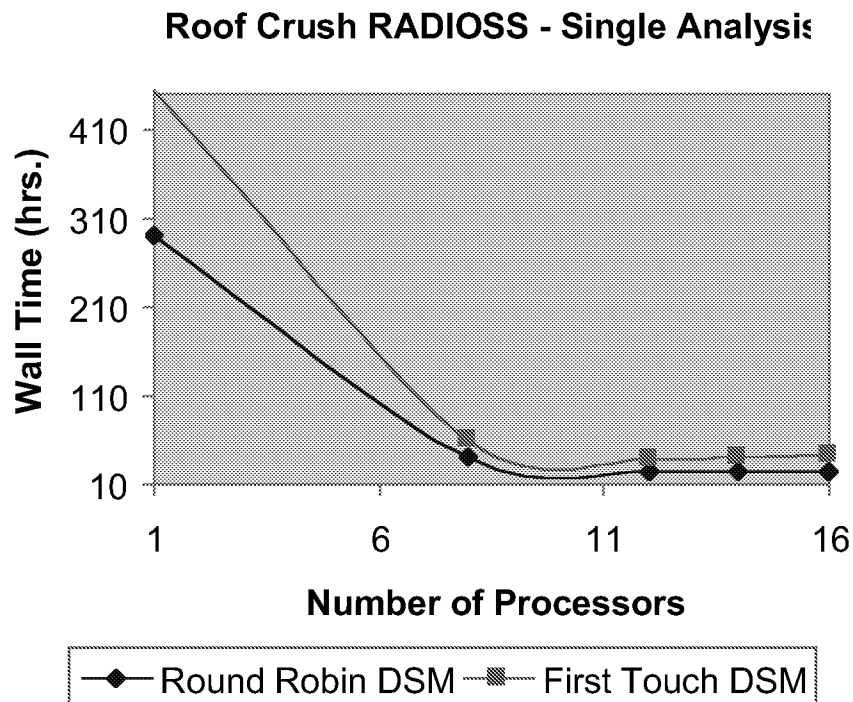


Figure 3.4.3-2: Wall time (hrs.) for a single RADIOSS analysis with variable number of processors

3.4.4 MDO Problem Results:

The MDO problem results in terms of the design variable and design response function values are documented in Table 3.4.4-1. The accuracy of the approximation models are tabulated in Table 3.4.4-2 and the final design deformed vehicle shape is shown in Figure 3.4.4-1.

As seen from Table 3.4.4-1, the initial design is an infeasible design with NVH discipline Static Torsion constraint violations of over 10%. The final design is a feasible design with a weight reduction compared to the initial design by 15 kg.

Number	Attribute Name	Initial Design	Cycle 1 (N=3)	Cycle 2 (N=2)	Lower Bound	Upper Bound
NVH DESIGN VARIABLES						
1	Rear floor panel	0.76	1.0	1.0	0.5	1.0
2	Rear floor cross m	1.4	2.0	2.0	0.8	2.0
3	Front floor pan	0.76	0.5	0.5	0.5	1.0
4	Front floor inner	1.07	1.0923	1.1926	0.5	1.5
5	Jacking/towing	0.8	1.5	1.5	0.5	1.5
6	Quarter panel	0.8	0.8876	0.8876	0.5	1.5
7	Backlite glass	3.8	2.6	2.6	2.6	5.0
8	Rear tire cover	0.75	1.0	1.0	0.5	1.0
9	Shotgun	1.22	1.3681	1.2643	0.9	1.5
10	Radiator support	0.76	0.5	0.5	0.5	1.0
11	K-7401	1073.3	1070.11	1193.41	750.	1395.
12	K-7402	366.9	477.95	478.0	256.	478.
13	K-7403	2733.6	2733.37	2734.24	1912.	3554.
14	K-7501	1424.5	1417.75	1438.51	1000.	1850.
15	K-7502	487.0	484.41	629.97	340.	630.
16	K-7503	3628.3	3627.95	3632.52	2540.	5090.
17	K-7601	1521.0	1518.30	1914.19	1065.	1977.
18	K-7602	520.0	513.96	675.0	365.	675.
19	K-7603	3874.0	3873.39	3886.59	2710.	5035.
COMMON DESIGN VARIABLES TO NVH & Crash						
20	Windshield	3.8	2.6	2.6	2.6	5.0
21	Roof Panel	0.7	0.4	0.4	0.4	1.0
22	Roof rail	0.8	0.6	0.6	0.6	1.2
23	Roof Cross Member F	0.8	0.6	0.6	0.6	1.2
24	Roof Cross Member R	0.7	0.5	0.5	0.5	1.1
25	A Pillar	0.8	1.0998	1.0971	0.5	1.1
26	B Pillar 1	0.8	0.7944	0.7788	0.4	1.0
27	B Pillar 2	0.8	0.5	0.5	0.5	1.1
28	B Pillar 3	1.35	0.9	0.9	0.9	1.5
29	C Pillar	0.8	0.5	0.5	0.5	1.1
Crash DESIGN VARIABLES						
30	Front door	0.7	0.4	0.4	0.4	1.0
31	Front door inner	0.7	0.4	0.4	0.4	1.0
32	Rear door	1.0	0.7	0.7	0.7	1.3
33	A Pillar 500071	0.207	0.345	0.192	0.192	0.345
34	A Pillar 500061	0.207	0.345	0.345	0.192	0.345
35	A Pillar 6182	0.207	0.345	0.192	0.192	0.345

36	B Pillar 500031	0.207	0.345	0.345	0.192	0.345
37	B Pillar 500001	0.345	0.192	0.345	0.192	0.345
38	F door inner 6192	0.207	0.192	0.192	0.192	0.345
39	F door inner 6194	0.207	0.345	0.345	0.192	0.345
NVH & Crash OUTPUTS – MDO PROBLEM OBJECTIVE AND CONSTRAINT RESPONSES						
1	NVH Weight	282.44	282.70	282.53	Objective	Minimize
2	Crash Weight	(A) 1255.65	1240.3	1240.2	Objective	Minimize
3	Mode 3 (Hz)	26.65	29.32	29.32	26.65	29.32
4	Static Torsion-Z disp: N99901 (mm)	3.67 (Violated)	3.29	3.29	None	3.3
5	Static Torsion-Z disp: N99902 (mm)	-3.68 (Violated)	-3.31	-3.31	-3.3	None
6	Static Bending-Z disp: Max. of 6 nodes (mm)	-0.97	-0.97	-0.935	None	1.2
7	Crash: NF – Normal reaction at I/F 2 (kN)	(A) 34.69	28.82 at t=27.28 msec	29.43 at t = 28.96msec	24.0	None
8	Internal Energy	(A) 3015.79	2331.7	2400.97	None	None

Table 3.4.4-1: Car MDO Problem Results

In Table 3.4.4-1, (A) refers to Approximate value based on the polynomial crash response surface model generated using 26 detailed RADIOSS analyses and (N) refers to the number of NVH approximation model updates within each outer cycle.

Response Name	Response Values		% Error between Actual & Approximate Values
	Actual	Approximate	
Weight (kg)	1522.73	1522.69	0.0
Mode 3 Frequency (hz)	29.32	29.32	0.0
Static Torsion (mm)	3.29	3.30	0.3
Static Bending (mm)	-0.935	-0.895	4.3
Crash Normal Force (kN)	29.43	30.57	3.9
Internal Energy	2400.97	2617.9	9.0

Table 3.4.4-2: NVH and Crash Approximation Model Errors (Cycle 2)

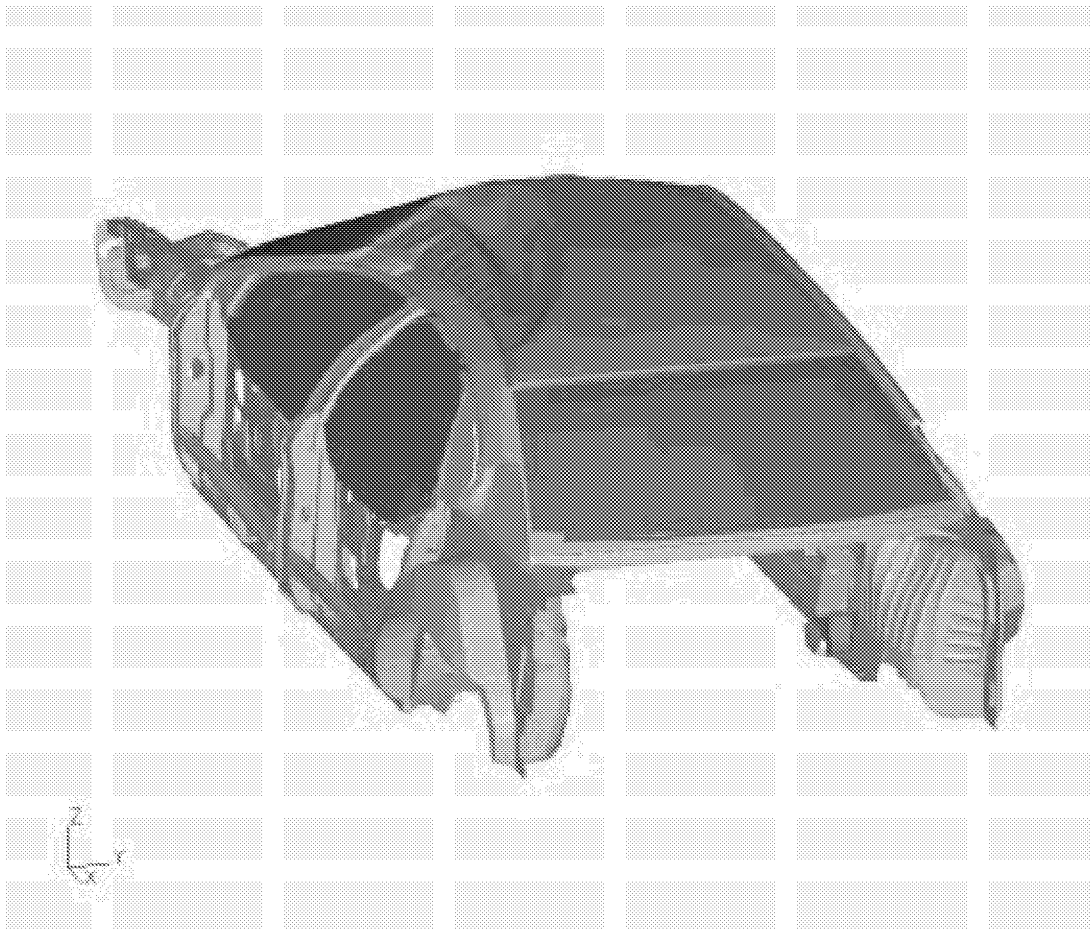


Figure 3.4.4-3: Deformed shape after roof impact corresponding to final design (cycle 2)

3.5 Summary

A new MDO method, BLISS, and two different variants of the method, BLISS/RS and BLISS/S, have been implemented using iSIGHT's scripting language and evaluated in this report on multidisciplinary problems. All of these methods are based on decomposing a modular system optimization into several subtasks optimization, that may be executed concurrently, and the system optimization that coordinates the subtasks optimization.

Detailed summary of the different methods are provided in the previous sections.

Interpreting the results, especially the work comparisons with A-i-O method, one should remember that the underlying analyses were simple, typical of the conceptual design stage. One expects that the cost of the system analysis (or the full FEA with BLISS/S) relative to the component analysis will increase as the design moved to the preliminary and detailed stages, hence the metric based on the number of the system analysis is likely to dominate.

The BLISS method and its variants are well suited for exploiting the concurrent processing capabilities in a multiprocessor machine. Several steps, including the local sensitivity analysis, local optimization, response surfaces construction and updates are all ideally suited for concurrent processing. Needless to mention, such algorithms that can effectively exploit the concurrent processing capabilities of the compute servers will be a key requirement for solving large-scale industrial design problems, such as the automotive vehicle problem detailed in Section 3.4.

4.0 References

Alexandrov, N., and Kodiyalam, S.: *Initial Results of An MDO Method Evaluation Study*, Proceedings, AIAA/USAF/NASA/ISSMO 7th Symposium on Multidisciplinary Analysis and Optimization, (held at St. Louis, MO), AIAA, Washington, D.C., September 1998. AIAA Conference Paper No. AIAA-98-4884.

Balling, R.J.; and Sobieszczanski-Sobieski, J.: An algorithm for solving the system-level problem in multilevel optimization, NASA CR-195015, December 1994.

Barthelemy, J.-F; and Sobieszczanski-Sobieski, J.: Optimum Sensitivity Derivatives of Objective Functions in Nonlinear Programming. AIAA Journal, Vol. 21, No. 6, pp. 913–915, June 1983.

Golovidov, O., Kodiyalam, S., Marineau, P. et al., “Flexible Implementation of Approximation Concepts in an MDO Framework,” Proceedings, 7th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization, AIAA, St. Louis, Missouri, September 1998. AIAA Paper No. 98-4959.

Kodiyalam, S.: Evaluation of Methods for Multidisciplinary Design Optimization (MDO), Phase I, NASA CR-1998-208716.

Kodiyalam, S. and Sobieszczanski-Sobieski, J.: Bi-Level Integrated System Synthesis with Response Surfaces, Proceedings, AIAA/ASME/ASCE/AHS/ASC 40th Structures, Structural Dynamics and Materials Conference, (held at St. Louis, MO), AIAA, April 1999. AIAA Conference Paper No. AIAA-99-1306. (To appear in AIAA Journal, August, 2000)

iSIGHT Designers and Developers Manual, version 4.0, Engineous Software Inc., Morrisville, North Carolina, 1998.

Renaud, J.: An Optimization Strategy for Multidisciplinary Systems Design, International Conference on Engineering Design, August 1993.

Sobieszczanski-Sobieski, J.; Agte, J.; and Sandusky, Jr., R.: Bi-level Integrated System Synthesis (BLISS), AIAA 98-4916, 7th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization, September 2-4, 1998, St. Louis, MO.

Sobieszczanski-Sobieski, J. and Kodiyalam, S.: BLISS/S: A New Method for Two-Level Structural Optimization, Proceedings, AIAA/ASME/ASCE/AHS/ASC 40th Structures, Structural Dynamics and Materials Conference, (held at St. Louis, MO), AIAA, April 1999. AIAA Conference Paper No. AIAA-99-1345. (Submitted to Structural Optimization Journal)

Sobieszczanski-Sobieski, J., Kodiyalam, S., and Yang, R-J.: Optimization of Car Body under Constraints of NVH and Crash, Submitted for Presentation at the AIAA/ASME/ASCE/AHS/ASC 41st Structures, Structural Dynamics and Materials Conference, (Atlanta, GA), AIAA, April 2000.

REPORT DOCUMENTATION PAGE			Form Approved OMB No. 0704-0188	
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.				
1. AGENCY USE ONLY (Leave blank)		2. REPORT DATE November 2000		3. REPORT TYPE AND DATES COVERED Contractor Report
4. TITLE AND SUBTITLE Evaluation of Methods for Multidisciplinary Design Optimization (MDO), Part II			5. FUNDING NUMBERS C NAS5-96012 WU 509-10-31-03	
6. AUTHOR(S) Srinivas Kodiyalam and Charles Yuan				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Engineous Software, Inc. 1800 Perimeter Park West Morrisville, NC 27560			8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) National Aeronautics and Space Administration Langley Research Center Hampton, VA 23681-2199			10. SPONSORING/MONITORING AGENCY REPORT NUMBER NASA/CR-2000-210313	
11. SUPPLEMENTARY NOTES This report was prepared for Langley under NASA Contract NAS5-96012, Langley Purchase Order L-9048. Langley Technical Monitor: Jaroslaw Sobieski				
12a. DISTRIBUTION/AVAILABILITY STATEMENT Unclassified-Unlimited Subject Category 64 Distribution: Nonstandard Availability: NASA CASI (301) 621-0390			12b. DISTRIBUTION CODE	
13. ABSTRACT (Maximum 200 words) A new MDO method, BLISS, and two different variants of the method, BLISS/RS and BLISS/S, have been implemented using iSIGHT's scripting language and evaluated in this report on multidisciplinary problems. All of these methods are based on decomposing a modular system optimization system into several subtasks optimization, that may be executed concurrently, and the system optimization that coordinates the subtasks optimization. The BLISS method and its variants are well suited for exploiting the concurrent processing capabilities in a multiprocessor machine. Several steps, including the local sensitivity analysis, local optimization, response surfaces construction and updates are all ideally suited for concurrent processing. Needless to mention, such algorithms that can effectively exploit the concurrent processing capabilities of the compute servers will be a key requirement for solving large-scale industrial design problems, such as the automotive vehicle problem detailed in Section 3.4.				
14. SUBJECT TERMS Parallel processing; Optimization by decomposition; Engineering systems design			15. NUMBER OF PAGES 51	
			16. PRICE CODE A04	
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT UL	